UNIVERSIDAD CARLOS III DE MADRID
ESCUELA POLITÉCNICA SUPERIOR

INGENIERÍA TÉCNICA EN INFORMÁTICA DE GESTIÓN

PROYECTO FIN DE CARRERA

MODELADO MULTIDIMENSIONAL: ANALISIS DE MECANISMOS DE EXTRACCIÓN DEL CONOCIMIENTO A PARTIR DE ESQUEMAS EER

Autora: Ana Esther Ramiro Bonilla
Tutora: Elena Castro Galán
Directora: Jessica Rivero Espinosa

Mayo, 2012

Agradecimientos

No existen palabras de agradecimiento para todas las personas que me han apoyado en este camino.

A mi tutora y directora mis más sincero agradecimiento por la dedicación y la ayuda prestada.

A mis tíos y amigos por animarme a continuar en la que ha sido una de las mejores experiencias, aunque, en ocasiones, el punto de vista fuera otro.

A todo CMC por haber sido tan estupendos compañeros, y en especial a esas personas que organizan al mejor equipo día a día y han hecho posible el desarrollo de este proyecto.

Muchísimas gracias a todos.

Índice de contenidos

Índice de Figuras

Índice de Tablas

Acrónimos y definiciones

- BI: *Business Intelligence*.
- E/R: *Entidad/Relación*.
- EER: *Entended Entity-Relationship*.
- SGBD: *Sistema Gestor de Base de Datos*.
- SGBDR: *Sistema Gestor de Base de Datos Relacional*.
- SGBDM: *Sistema Gestor de Base de Datos Multidimensionales*.
- PWC: *Informática PowerCenter*.
- ODUNI: *Origen de Datos de UNIversidades*.
- UNICUBE: *CUBO de UNIversidades*

Capítulo 1

En este primer capítulo se ofrece una introducción del que será el tema a desarrollar a lo largo del documento. Se situará el proyecto en un contexto y se introducirán los términos más generales que servirán de base para el desarrollo del proyecto.

En este capítulo introductorio también se describirá la estructura del proyecto con un breve resumen de cada apartado.

1. INTRODUCCIÓN

Los sistemas de información tradicionales, como las aplicaciones personalizadas o los programas de gestión, presentan un diseño capaz de adaptarse, de mejor o peor manera, a los requisitos de una empresa. Sin embargo, presentan una estructura muy rígida que, si no imposibilita, hace muy complicado extraer el conocimiento almacenado en las diferentes bases de datos que la empresa maneja y cuyas referencias son actualizadas diariamente.

El *Business Intelligence* nace de la necesidad de extraer el conocimiento que puedan aportar los datos almacenados apoyándose en una serie de herramientas que permitan superar las limitaciones hasta ahora impuestas por los sistemas tradicionales. A través de procesos de carga, extracción, depuración y transformación de la información se almacenarán los datos finales con una organización que permita generar un conocimiento que apoye a la toma de decisiones de la empresa. Este almacenamiento se realiza, generalmente, en un *Data Warehouse* corporativo.

Los sistemas tradicionales no son herramientas alternativas a sistemas de *Business Intelligence*, sino complementarias a éstos. Los sistemas *Business Intelligence* necesitan de la información recogida por ellos.

Se denomina *Business Intelligence* al conjunto de estrategias, tecnologías y aplicaciones orientadas a transformar y estructurar los datos de que dispone una empresa para explotarlos directamente y analizarlos, llegando así al conocimiento necesario que dé soporte a la toma de decisiones de la empresa.

Este proyecto final de carrera ahondará en estos sistemas de *Business Intelligence*, en adelante *BI*, y en su principal fuente de datos, un *Data Warehouse* (DWH).

1.1. Contexto

Usualmente la información de negocio está repartida de forma inconexa, y en muchos casos incompleta, para las diferentes partes de la estructura de la empresa que la precisa, siendo los usuarios los que han de interpretar los datos.

Es frecuente que cuando el usuario final busca recuperar la información de un origen de datos, resulte complicado comprender lo que contienen estos orígenes, pues se diseñan desde el punto de vista de quién los crea, los programadores, y no desde el de los usuarios. Esto hace que la información que resulta de interés pueda estar distribuida en diferentes tablas, esquemas y bases de datos (BD), por lo que se debe conocer el detalle de toda la arquitectura, además del lenguaje de programación (SQL). Incluso, la información puede recogerse de servicios y aplicaciones web y/o archivos.

Este origen, con frecuencia tan heterogéneo, presenta además demasiado nivel de detalle en algunos casos, haciendo necesarias consultas que agreguen la información, resumiendo los datos para poder estudiarlos. Con el aumento de los datos, obtener estos resúmenes para poder tomar decisiones corporativas puede suponer un imposible para el usuario.

Esto supone un problema, ya que las empresas toman decisiones constantemente, pudiendo éstas llevarlas al éxito o abocarlas al fracaso. Debido a ello, la capacidad de tomar decisiones precisas y rápidas se ha convertido en una meta de los diferentes negocios, que buscan explotar la información que tienen a su disposición para conquistar el mercado.

Una encuesta realizada por IBM [IBM, 2011] a 225 líderes empresariales descubrió lo siguiente:

8 de cada 10 directivos encuestados toman decisiones a partir de información incompleta o en la que no confían plenamente**.**	**1 de cada 2** directivos encuestados no tiene acceso a la información que necesitan para hacer su trabajo.
3 de cada 4 directivos afirman que disponer de información predictiva les permitiría tomar mejores decisiones.	**2 de cada 3** se encuentran en una fase inicial en sus proyectos para conseguir información predictiva.

Para tomar las decisiones más acertadas se debe tener una información completa y veraz, sin incongruencias. Unos datos en los que, la persona que finalmente tome una determinación, pueda confiar.

Se debe tener la posibilidad de acceder a unos datos que lleven a un conocimiento de la situación, y con los que se pueda observar una tendencia que permita predecir escenarios.

Se habla de "*partir de datos*" y "*llegar a conocimiento*", para llegar de un punto a otro es necesaria la información. Basándose en las definiciones de Davenport y Pursak [Davenport, T.; Prusak, L., 1998] hay tres conceptos diferenciados: *dato*, *información* y *conocimiento*.

- Un hecho real presenta factores objetivos, denominados ***datos***, que poseen un conjunto discreto de valores. Un *dato* por sí mismo apenas tiene relevancia, sin un contexto no es explicativo.

  Los *datos* son almacenados mediante tecnologías varias, de estas tecnologías se evalúa la capacidad de almacenamiento y procesamiento, el coste asociado o la velocidad de procesado de los datos.

Los *datos* simplemente constituyen la realidad, o una parte de esta, y en ningún caso son interpretaciones o pueden formar juicios de valor, no dicen qué *dato* es el importante y cual el menos significativo. La toma de decisiones se basa en los *datos*, pero ellos no indican directamente cual es la decisión que debe tomar la directiva de una organización o el departamento de una empresa, son la base para la creación de la información.

Toda organización precisa de los *datos*, y algunos sectores son dependientes de ellos, como es el caso de las finanzas y los seguros o las agencias gubernamentales en dónde existen millones de transacciones. En estos casos la gestión óptima de los *datos* es crítica.

Con esto no se debe concluir que a mayor cantidad de *datos* almacenados mejor para la compañía. Las empresas llegan a acumular *datos* sin sentido, por lo que para gran parte de ellas, contar con muchos *datos* no es lo mejor. Esta operativa no tiene sentido porque, en primer lugar, los *datos* en sí mismos no tienen sentido, por lo que si no se tratan y estudian, de nada sirve su almacenaje y se convierten en un inconveniente: se pierde capacidad de procesado y resultan costosos de mantener, y en segundo lugar demasiados *datos* hacen más complicado identificar los que son relevantes.

- La ***información*** es un mensaje, ya sea en formato escrito, audible o visible, tiene significado en sí misma, está organizada con algún propósito, y es capaz de cambiar la percepción del receptor al que informa. Por tanto, es el receptor de la *información* quién determina si ésta realmente le informa, y no el emisor del mensaje. Un documento puede resultar muy informativo para quién lo escribe, emisor, y a la vez constituir ruido para el receptor.

  Cuando el creador del mensaje añade significado a los *datos*, éstos pasan a ser *información*. La *información* es por tanto la comunicación de conocimiento. Existen métodos para transformar los *datos* en *información*, métodos que añaden valor en diferentes sentidos:

  - Contextualizando: se conoce el propósito de la generación de esos *datos*.
  - Categorizando: se conocen los sistemas de medida necesarios para la interpretación.
  - Calculando: se pueden procesar los *datos* de forma matemática o estadística.
  - Corrigiendo: se suprimen los errores de los *datos*.
  - Condensando: se pueden resumir, agregar, los *datos* de forma más concisa.

  Resulta complicado que las tecnologías ayuden a analizar el contexto de la *información*, sin embargo tratan los *datos* transformándolos en *información*. Es importante no confundir la tecnología, que es el medio, con la *información*, que es el mensaje. Tener acceso a más tecnologías de la información no implica mejorar el nivel de información

- El ***conocimiento*** intuitivamente parece algo más complejo y amplio. Para Davenport y Pursak [Davenport, T.; Prusak, L., 1998]. "*Es una mezcla de experiencia, valores, información y "saber hacer" que sirve como marco para la incorporación de nuevas experiencias e información, y es útil para la acción. Se origina y aplica en la mente de los conocedores. En las organizaciones con frecuencia no sólo se encuentra dentro de documentos o almacenes de datos, sino que también está en rutinas organizativas, procesos, prácticas, y normas*."

  Al igual que la *información* deriva de los *datos*, el *conocimiento* deriva de la *información* y se encuentra en las personas, en los grupos de conocimiento o en rutinas organizativas. Para derivar en *conocimiento* existen acciones como:

  - Comparar con otros casos.
  - Predecir consecuencias.
  - Buscar conexiones.
  - Entablar conversaciones con otros individuos portadores de conocimiento.

Un sistema de *BI* utiliza los datos para ofrecer una información con la que llegar a un conocimiento que permita tomar decisiones. Estos datos de diferente procedencia se unificarán y almacenarán, para servir de origen de información a un sistema de *BI*, en un *DWH*, que es en esencia, una colección de datos orientada a un ámbito, integrados, no volátiles e historiados y organizados.

1.2. Objetivos

Con el presente proyecto fin de carrera se pretende comprender un sistema de *Business Intelligence*, exponiendo un ejemplo con algunas de las herramientas comerciales que actualmente usan las empresas.

Según documentos publicados por Gartner Research (2011). "*El objetivo del Business Intelligence es eliminar las conjeturas y la ignorancia en los ambientes empresariales, aprovechando los vastos volúmenes de datos cuantitativos que las empresas recolectan todos los días en sus diversas aplicaciones corporativas."*

Siguiendo la definición, existen multitud de datos objeto de estudio. Por ello, el proyecto abordará la creación de una base de datos relacional, basada en un esquema *Entidad/Relación*, *E/R* en adelante, y el proceso de transformación de este esquema, mediante un proceso de *Extracción Transformación y Carga* (*ETL*, de las siglas en inglés *Extrac, Transfomation and Load*), en un *esquema de estrella* y otro en *copo de nieve*.

Los esquemas transformados conformarán los *Datamart* que definen un *DWH*, que es el origen de datos necesarios para llegar al conocimiento que un sistema de *BI* proporciona.

Para una descripción más concreta de un proyecto de *BI*, además de la creación y transformación de los modelos. El modelo *E/R* inicial se completará en los sistemas, con información ficticia, a través de un proceso *ETL* y se desarrollará un *cubo multidimensional* (*cubo*), que permita la explotación de la información almacenada en el *esquema en estrella* transformado.

1.3. Estructura del Documento

A continuación, un preludio del contenido del proyecto, estructurado en diez capítulos y una serie de anexos:

Capítulo 1: Introducción

Introducción al *BI*, descripción de los conceptos *dato*, *información* y *conocimiento*, base de los temas a desarrollar en los capítulos sucesivos. Situación en el contexto actual y definición de objetivos del proyecto.

Forma parte de este capítulo esta descripción de la estructura del documento.

Capítulo 2: Estado del arte

Desarrollo más minucioso de la arquitectura de un sistema *BI*. Se tratan las entradas, la construcción y la explotación de un sistema de estas características y los productos resultantes como puntos principales.

En este capítulo, cabe destacar la construcción del sistema de *BI*, cuya parte fundamental es el objetivo principal del proyecto, la transformación entre modelos de almacenamiento de datos, mediante un proceso *ETL*.

El *Cubo multidimensional*, como método de explotación de la información más adecuado, es otro de los objetivos a desarrollar en el proyecto descrito en un apartado del capítulo.

Capítulo 3: Estudio de la viabilidad y gestión del proyecto

Estudio de las herramientas Informática Power Center e IBM Cognos 8, empleadas en el tipo de desarrollos estudiados. Análisis de sus componentes, funcionalidades de los mismos y operadores definidos en cada uno de ellos.

Presentación de alternativas a las herramientas seleccionadas y un plan del proyecto con la planificación temporal y presupuestaria del mismo.

Capítulo 4: Análisis y diseño del sistema

Análisis del sistema planteado y diseño del mismo, haciendo uso de la metodología establecida para el modelado *E/R, esquemas en estrella* y *esquemas en copo de nieve*.

Estudio de automatización del proceso de conversión entre esquemas relacionales y esquemas relacionales para la representación de datos dimensionales.

Capítulo 5: Implementación del sistema

Puesta en práctica de lo descrito y diseñado en los apartados previos. Implementación de una base de datos relacional bajo un modelo *E/R* y conversión de ésta en un *esquema en estrella*. Posterior lectura del sistema para la generación de un *cubo* funcional.

Capítulo 6: Experimentación y resultados

Pruebas y testeo del sistema construido, a fin de asegurar la integridad de la información a lo largo del proceso de conversión entre modelos y pruebas de análisis con el *cubo* construido.

Capítulo 7: Conclusiones

Corolario del ejercicio llevado a cabo.

Capítulo 8: Líneas futuras

Tratamiento de posibles evoluciones del proyecto, que amplíen el alcance concebido en esta primera edición.

Bibliografía

Catálogo de las referencias utilizadas para el desarrollo del proyecto y que se mencionan a lo largo de éste.

Anexos

En los anexos aparece documentación relacionada con el proyecto que puede resultar de interés: Manual de usuario del *cubo* y configuración de una base de datos ODBC.

Capítulo 2

En este capítulo se profundizará en un sistema de *BI*, se definirá la arquitectura general de una solución de *BI*, cuáles son sus fuentes de información o entradas principales, cómo se construye un sistema de estas características, cómo se explota la información recogida y estructurada, y los productos que se obtienen.

2. ESTADO DEL ARTE

En una solución de *BI* existen una serie de niveles desde que se obtienen los datos hasta su presentación para la extracción de conocimiento.

Los datos de los que dispone una organización tienen su origen en diferentes fuentes, están aislados, presentan diferentes formatos, etc. Todos esos datos sufren una transformación, para garantizar la coherencia y consistencia de la información, y son cargados en los diferentes *Datamart* que componen un *DWH* corporativo, mediante un proceso de *ETL*.

Desde el *DWH* se alimentan herramientas que permiten operar con facilidad sobre gran cantidad de datos. Entre ellas, la más destacada son los *Cubos OLAP* o *Cubos multidimensionales*. Además, esta información puede ser gestionada para ofrecer informes especializados en el análisis de datos.

La arquitectura general se puede observar en la Ilustración 1 y será definida en este capítulo.

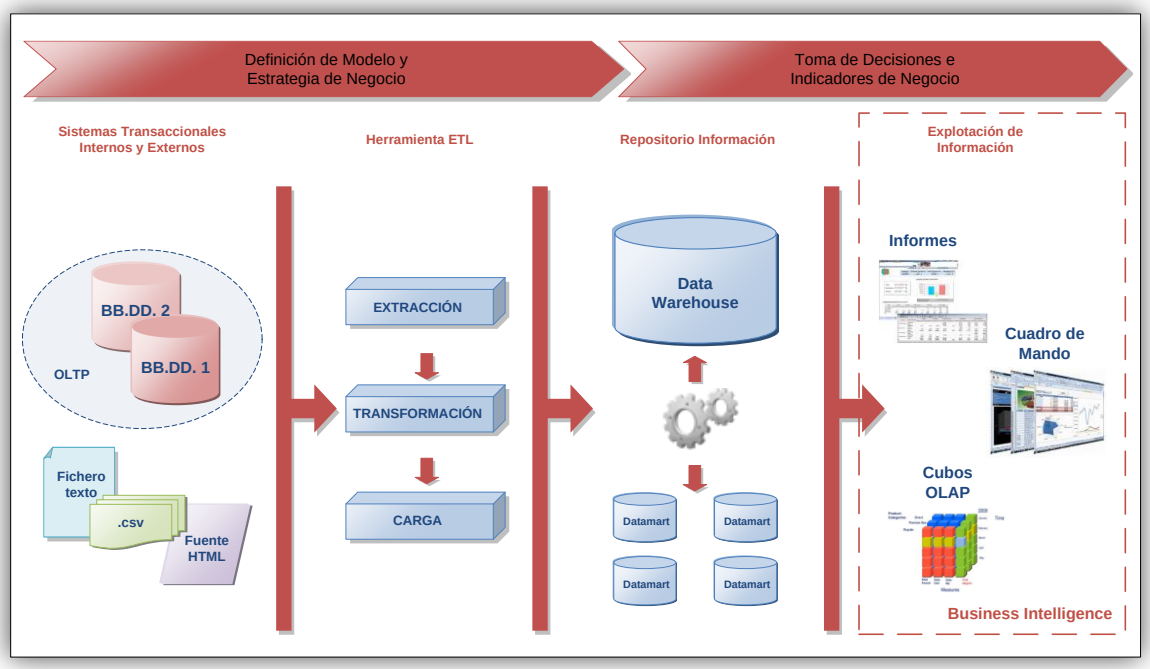


Ilustración 1. Arquitectura BI

2.1. Entradas de un sistema de *BI*

Un sistema de *BI* maneja gran cantidad de datos de diversa índole y origen dispar. Todas estas referencias necesarias deben unificarse para formar una entrada única, homogénea y consistente. Esta entrada es, generalmente, un *DWH* alimentado por sistemas operacionales (relacionados con las BD OLTP) y que sirve de origen para los sistemas analíticos en línea (relacionado con las BD *OLAP*).

2.1.1. Base de datos *OLTP*: On-Line Transactional Processing

Se trata de BD orientadas al proceso de transacciones tales como inserciones, modificaciones y borrados. Es típico de las *BD operacionales.*

- Acceso optimizado a los datos en las tareas de lectura y escritura, las más frecuentes.
- Los datos están estructurados según el nivel de aplicación.
- No tiene porque existir la uniformidad en los datos de diferentes áreas.
- Suelen tratarse datos recientes y actuales, no es común un histórico.

2.1.2. Base de datos OLAP: On-Line Analytical Processing

Se trata de BD orientadas a un procesamiento analítico de los datos. El proceso analítico precisa de lecturas de grandes cantidades de datos con el fin de extraer información de interés, para elaborar informes, o extraer tendencias y patrones.

Es el sistema propio de los *Datamarts*, bases de datos departamentales, específicas para el almacenamiento de los datos de un área concreta.

Por lo general, un sistema *OLAP* obtiene la información de sistemas operacionales (típicos *OLTP*) existentes a través de procesos *ETL*. Generalizando, podría decirse que los sistemas OLTP proporcionan los datos mientras que los OLAP ayudan a analizarlos.

Las características más destacadas de estos sistemas son las que siguen:

- Lo más habitual es la realización de consultas, predominando por tanto las operaciones de lectura, y siendo las menos las operaciones de inserción, modificación o borrado.
- Los datos están estructurados según el departamento.

- Como consecuencia de la estructuración, los datos son uniformes.
- Suele tratarse un historial de datos importante, de dos a cinco años.

En estos sistemas, ligados a los *cubos*, que son estructuras multidimensionales formadas por unos o varios *Datamarts*, donde se definen dimensiones y jerarquías, es decir, se trata de objetos *OLAP* que se configuran en el nivel de base de datos. Estos objetos son independientes de los *cubos* que nutren de información, por lo que pueden utilizarse en unos o más *cubos* diferentes. Más adelante, en el punto 2.3.3. Cubos OLAP, se desarrolla el concepto de *cubo*.

2.1.3. Data Warehouse

Inmon en una de sus obras, "Building the Data Warehouse", [Inmon W.H., 1992] definió: "*El Data Warehouse es una colección de datos orientados al tema, integrados, no volátiles e historiados, organizados para el apoyo de un proceso de ayuda a la toma de decisiones*".

Así, un *DWH* es una colección de datos orientada a un ámbito y que cumple las siguientes características:

- Es ***integrado***, existe una estructura consistente en la que se alojan los datos, por lo que las inconsistencias entre los propios datos, que llegan a la estructura de diferentes sistemas operacionales, deben ser eliminadas.

  Las inconsistencias se suelen deber a que las diferentes fuentes de información que nutren el almacén de datos no están definidas a través de un estándar, los diseñadores y programadores que crearon las diferentes bases de datos y aplicaciones, orígenes de los datos, lo hicieron de formas diferentes, creando modelos inconsistentes e incompatibles entre sí.

  La estructura de un *DWH* suele estar jerarquizada en distintos niveles de detalle que permiten un mejor uso de la información por los distintos usuarios, quienes no requieren la misma cantidad de detalles.

- Es ***temático***, los datos se organizan por temas para lograr un mejor entendimiento por parte del usuario final. Sólo la información que es necesaria para generar conocimiento es integrada en el sistema. Así, los datos relativos al personal de una empresa se recogerían en una única tabla, independientemente del entorno operacional del que procedan.

- Es ***no volátil***, los datos que existen en un sistema de este tipo se almacenaron para la consulta, por lo que las modificaciones consisten en incorporaciones de los últimos valores sin modificación de la información que ya existía.

La información es útil para el análisis y la toma de decisiones solo cuando es estable, por lo que esta característica es muy valorada.

- Es ***histórico***, consecuencia de lo anterior es la variable tiempo, presente en todo el *DWH*. Los datos de los sistemas operacionales llegan representado un momento concreto de la realidad que describen. Cuando esta cambia la información se añade con la variable tiempo modificada (información no volátil). De esta forma los datos sirven para analizar tendencias a lo largo del tiempo.

Además de esto, un *DWH* contiene metadatos, esto es, información sobre los propios datos tal como la procedencia del dato, su forma de cálculo o su frecuencia de modificación.

El objetivo de los metadatos no es otro que ofrecer información adicional al usuario, y ser útil a los responsables técnicos que administran el almacén de datos y elaboran programas que extraen la información.

Un *DWH*, o almacén de datos, se nutre de toda la información de que dispone una empresa, ya sea externa o interna, y que puede llegar desde *bases de datos relacionales* (*BBDDR*), ficheros de texto o aplicaciones personalizadas. El proceso que carga el *DWH*, previa transformación de la información, es el proceso *ETL*. De este proceso *ETL* se hablará detalladamente más adelante en este capítulo.

Las contribuciones de un *DWH* no son pocas:

- Este cúmulo de datos es información integrada de todo el negocio, por lo que proporciona una gran herramienta para la toma de decisiones para cualquier área funcional del mismo.

- Optimiza la tecnología y economiza los recursos. Se eliminan las islas de información que puedan formar diferentes departamentos, centralizando la información en una única plataforma sólida y formando un centro de generación de informes.

- Las técnicas de análisis son más fácilmente aplicables. Así como también resulta más sencillo encontrar relaciones ocultas entre los datos de la compañía, relaciones que suponen un valor añadido al negocio.

- Como consecuencia de lo anterior, aumenta la competitividad de la empresa en el mercado.

- Proporciona un gran histórico de información, útil para aprender del pasado y predecir situaciones futuras. Es decir, hace más competitivos a los responsables de la toma de decisiones en un negocio, permite la toma de decisiones tácticas y estratégicas.

Sin embargo, existen inconvenientes en un *DWH*:

- Tiene un coste elevado. Una construcción adecuada requiere de un análisis profundo que consume gran cantidad de recursos, además de la propia implementación que implica personal cualificado y herramientas apropiadas.

- Los datos sensibles pueden ser un problema, pues pueden ser visibles para usuarios no autorizados.

- Los beneficios de un almacén de datos son visibles a medio-largo plazo, por lo que este inconveniente genera otro: la mayoría de usuarios de este sistema no confían en él al principio, son reacios al cambio.

Otro autor importante, en este sector de las bases de datos, es Ralph Kimball [Kimball R., 1998], quién define un almacén de datos como "*una copia de las transacciones de datos específicamente estructurada para la consulta y el análisis*", y fue también este autor quién, defendiendo una metodología de diseño ascendente para estos sistemas, determinó que un *Data Warehouse* no era sino "*la unión de todos los Datamarts de una entidad*".

De esta última definición surge el concepto de *Datamarts.* Un *DWH* está constituido por diferentes *Datamart*, éstas son bases de datos departamentales, especificas para el almacenamiento de los datos de un área concreta.

Los *Datamarts* representan la unidad básica de un *DWH*. En un sistema de *BBDDR*, un *Datamart*, físicamente, es un conjunto de tablas con una estructura específica para mejorar el proceso de consulta y análisis. Esta estructura se puede plantear bien como una base de datos *OLTP*, *On-Line Transactional Processing*, u *OLAP*, *On-Line Analytical Processing*. Como se describía en el punto anterior, esta última es la estructura más acertada para los *Datamarts*.

2.2. Construcción de un sistema de *BI*

La construcción de cada nuevo sistema *BI* pasa por los siguientes puntos:

(1) Identificar la información necesaria para dar soporte a la toma de decisiones

(2) Buscar la información identificada en los sistemas operacionales (OLTP)

(3) Recoger la información

(4) Integrar la información

(5) Cargar los diferentes Datamart

(6) Facilitar al usuario final la explotación de los Datamart

Este punto es de crucial importancia en los sistemas de *BI*, puesto que de esta construcción dependerá la calidad de los datos obtenidos y con ello la calidad del conocimiento adquirido. Además, una correcta construcción optimiza los tiempos de respuesta y los recursos de almacenamiento.

A continuación, se ahonda en cada una de las fases referenciadas.

(1) Identificar la información necesaria para dar soporte a la toma de decisiones

Dependiendo de las necesidades de quién recurre a un sistema de análisis de *BI*, la información útil será de uno u otro tipo. Se requiere de expertos en la materia que se analiza para poder identificar la información de interés.

Por ejemplo, para analizar la productividad de un operario de asistencias técnicas en una empresa de telefonía, se precisará de un experto en este sector, que establezca que se debe estudiar, y que será información útil las horas de trabajo diarias, el número de incidencias que recibe o el tiempo de resolución de cada una y no será de utilidad conocer el número de teléfono del operario o su comida favorita.

(2) Buscar la información identificada en los sistemas *OLTP*

Una vez se conoce la información que será relevante para el estudio y análisis de una situación, se debe localizar en los diferentes sistemas operacionales de los que dispone.

En ocasiones, puede no estar recogiéndose una información particular, establecida como necesaria en la fase anterior. En estos casos, debe estudiarse si la falta de esta es asumible, puesto que si se estableció como necesaria tiene repercusiones negativas en la calidad del análisis, o en caso de no serlo, donde se almacenará la información que se precisa.

(3) Recoger la información

Las fase de la (3) a la (5) conforman la denominada *ETL*, proceso de extracción transformación y carga de la información al que se ha hecho referencia en los apartados anteriores del proyecto.

Para la realización de un proceso como éste se pueden utilizar una gran variedad de tecnologías. Procesos desarrollados en Shell-script, C o SQL suponen la opción más básica y a la vez más complicada (todo se controla con la programación), pero un buen desarrollo puede aportar un gran rendimiento.

Existen herramientas comerciales que suelen tener una interfaz gráfica, lo que facilita y aporta una gran rapidez y facilidad a la hora de desarrollar el proceso *ETL*.

Es el caso de Informática PowerCenter, Oracle WareHouse Builder, Data Integrator (BO), ETI*Extract que liberan de muchos parámetros ya implementados como el procesamiento paralelo, el acceso a fuentes de datos heterogéneas, preparación de índices y caches, etc.

Independientemente de la tecnología elegida, la primera acción de esta etapa *ETL* es realizar una copia de las fuentes de datos en bruto (BD, ficheros, sistemas operacionales y otros orígenes). Debido a la volatilidad de los datos, para futuros usos, se guarda un duplicado. Esta copia en bruto suele hacerse en un área de la BD denominada "Staging Area".

Una vez se ha realizado la copia, se selecciona la información relevante que será extraída de los sistemas y transformada posteriormente. La información principal es la que se determinó y localizó en las fases anteriores, puntos (1) y (2), de construcción del sistema de *BI*.

(4) Integrar la información

La parte de la *ETL* más crítica es el proceso de transformación, se deben integrar y unificar los datos heterogéneos, agrupándolos de manera lógica según el conocimiento del negocio. Es muy posible que sea necesario generar nuevas claves primarias para los datos y trasformar unidades y formatos de los datos numéricos.

Durante este proceso se comprobará y arreglará la integridad de los datos, asegurando que todos cumplen con sus dependencias funcionales y rechazando los datos erróneos.

La información que llegó a esta fase del proceso estaba almacenada siguiendo un esquema *E/R* o incluso desestructurada, sin seguir esquemas, la integración de la información esta focalizada a conseguir una estructuración de los datos.

Los *esquemas en estrella* y en *esquemas en copo de nieve* son modelos dimensionales típicos en estos procesos. Ambos establecen unas *dimensiones* y unos *hechos*.

Una *dimensión* es una jerarquía de elementos que representan un concepto definido por el negocio de la empresa o por el usuario, por ejemplo, la dimensión de Tiempo y su jerarquía de Año-Trimestre-Mes-Día o la dimensión de Ubicación y su jerarquía País-Ciudad.

Un *hecho* es un dato, un valor numérico o textual que contiene la información de las operaciones del negocio de la empresa, por ejemplo, el número de horas trabajadas, el coste de una hora de trabajo o el número de clientes.

Esquema en estrella, es quizá el más sencillo y la representación más importante de los esquemas relacionales adaptados a la representación de datos multidimensionales.

El esquema define una serie de *tablas de dimensiones* y una serie de *tablas de hechos*. En las *tablas de dimensiones*, las columnas representan las jerarquías o niveles de la *dimensión*. Los *hechos* son los objetos analizados a través de las *dimensiones*, las columnas son los valores de las dimensiones y los *atributos de hecho*, aquellos atributos cuantitativos son las *medidas* que se analizan para ese *hecho*. En adelante, también se hará referencia a las *medidas* como *atributos de hecho* o *indicadores.*

El esquema dejaría la *tabla de hechos* en el centro y las *dimensiones* alrededor, dándole una forma estrellada que da pie al nombre de este tipo de esquema.

Este es un modelo des-normalizado, lo que favorece al tiempo de respuesta de las consultas que no precisan hacer uniones entre tablas, los datos pre-calculados ya se encuentran almacenados. Se presenta un modelo de datos fácil de comprender y modificar, pues existe un paralelismo entre su diseño y la forma en que los usuarios finales visualizan la información, lo que también hace más fácil el análisis.

Un ejemplo muy utilizado son las ventas, Ilustración 2. Esquema en estrella. En este caso, las ventas serían el *hecho* y sus *dimensiones*: tiempo, localización y producto. Una medida de ventas sería el número de unidades vendidas o el valor de la venta. En las *dimensiones* se encontrarían las jerarquías día, mes y año (dimensión tiempo), tienda, localidad y provincia (dimensión localización) y producto, familia y departamento (dimensión producto).

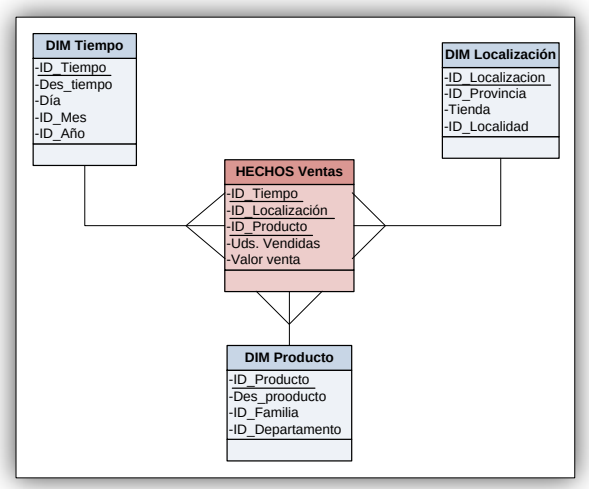


Ilustración 2. Esquema en estrella

Esquema en copo de nieve, más complejo que un esquema en forma de estrella, se trata de una extensión de éste.

Esta estructura se presenta cuando una o más *dimensiones* se implementan con más de una tabla de la BD. La finalidad es normalizar las tablas para evitar la redundancia de los datos, pero esto presenta un inconveniente muy importante: las *tablas de dimensiones* aumentan así como las relaciones entre ellas. Esto empeora el rendimiento, pues una consulta precisa de numerosas uniones (*JOINS*) que relacionen las tablas, lo que hace más pesada la operación.

Al tratarse de un modelo normalizado es más cercano al modelo *E/R* y también requiere de menos esfuerzo en el diseño.

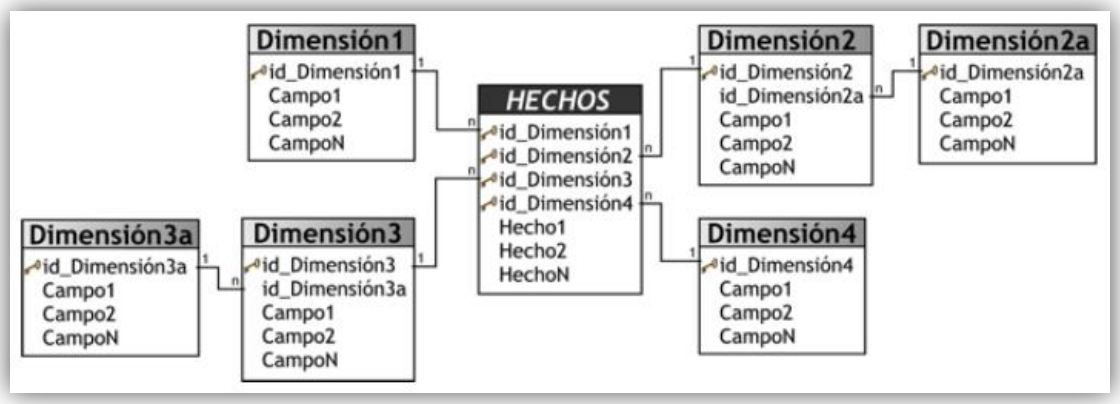


Ilustración 3. Esquema en copo de nieve

La principal diferencia que tienen estos dos esquemas básicos es que, en el segundo las *dimensiones* están normalizadas, lo que evita la redundancia de los datos, inevitable con la des-normalización de la información, y ahorra en espacio, si bien este ahorro no es significativo en la actualidad, donde el espacio en disco no supone una barrera en los recursos de una empresa.

Ambos tipos presentan ventajas y desventajas, pero la elección de uno u otro esquema suele determinarse por la herramienta que lo explote y el tipo de consultas que se realicen. No obstante, en un sistema donde el tiempo de respuesta sea crítico se optará por el *esquema en estrella*.

(5) <u>Cargar los diferentes *Datamart*</u>

Partiendo de los datos ya integrados se podrán cargar los distintos *Datamart*. El primer punto es la carga de las dimensiones necesarias definidas para un *Datamart* específico, el segundo paso es cargar los hechos.

Físicamente, los *Datamart* son un conjunto de tablas de la BB.DD. La estructura que guardan estas tablas y el tipo de datos elegido para cada campo se establece de forma que se mejore el proceso de consulta. Se basan en la des-normalización de la información, ya que en este punto los datos están integrados y validados y la normalización resulta innecesaria, liberando al *Sistema Gestor de Base de Datos Relacional (SGBDR)* de cualquier control sobre los datos (restricciones, claves primarias y externas, etc.)

Las *dimensiones* dependen de la estructura de tabla o tablas que forman los *Datamarts*, puede tratarse de *esquemas en estrella* o en *copo de nieve*.

La Ilustración 4 es un ejemplo donde se observa cómo organiza, desde un punto de vista lógico, un *Datamart* los datos disponibles.

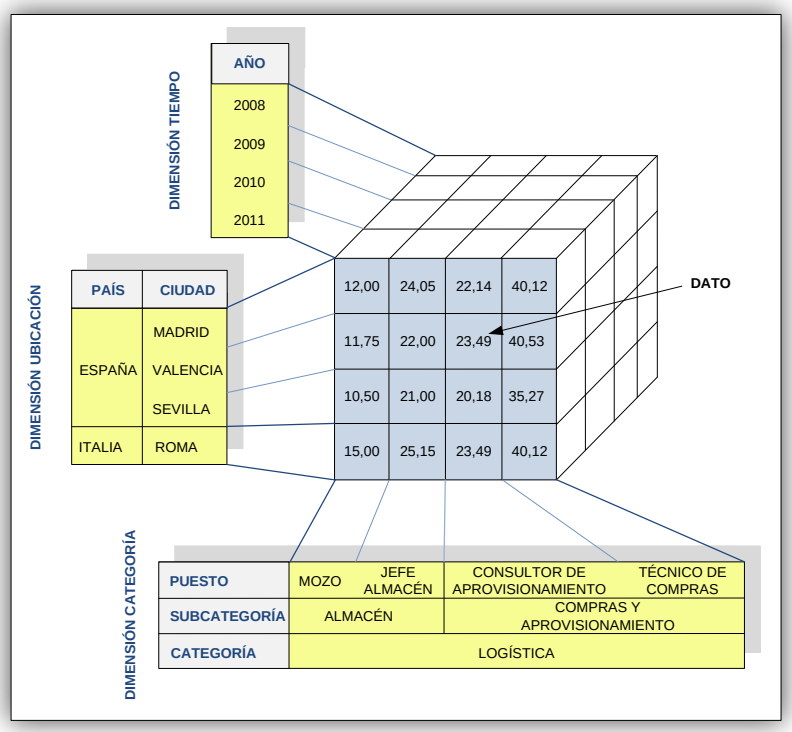


Ilustración 4. Ejemplo gráfico Datamart

En este *Datamart* sólo aparece el *indicador* "coste por hora trabajada". Fijando los datos de las *dimensiones* por una ubicación (país-ciudad), categoría (categoría-subcategoría-puesto) y tiempo (año), se obtendría el dato de más detalle, el *hecho*, una celda del *cubo*. Si por el contrario no se fijaran restricciones se sumaría el coste por hora de todo el territorio, independientemente de la categoría o el tiempo.

A pesar de mostrarse únicamente tres *dimensiones*, para la mejor comprensión del esquema, no existe restricción en este sentido.

Se observa que las *dimensiones* están jerarquizadas. No obstante, para fijarlas no es preciso llegar al escalón más bajo de la jerarquía que tengan definida. Así, las celdas del *cubo*, los *hechos*, muestran el coste por hora de las ciudades, porque se ha bajado a la jerarquía ciudad de la *dimensión* ubicación. Mientras, en la *dimensión* tiempo se muestra únicamente el año, por lo que el valor corresponde a la media de ese año (podría estipularse la suma, el mínimo…). Habitualmente esta última *dimensión* está jerarquizada en año-trimestre-mes-día.

Con el mismo ejemplo, podría fijarse la *dimensión* tiempo con la jerarquía mes y no haberse hecho desglose en la ubicación, mostrando la suma de las ciudades. Con este último cambio quedaría como muestra la Ilustración 5. En este ejemplo se ha supuesto que el coste por hora no ha variado en los meses de año 2011 (*dimensión* tiempo) para mostrar claramente que se trata de la suma de costes de las ciudades del país (*dimensión* ubicación) en abril de 2011.

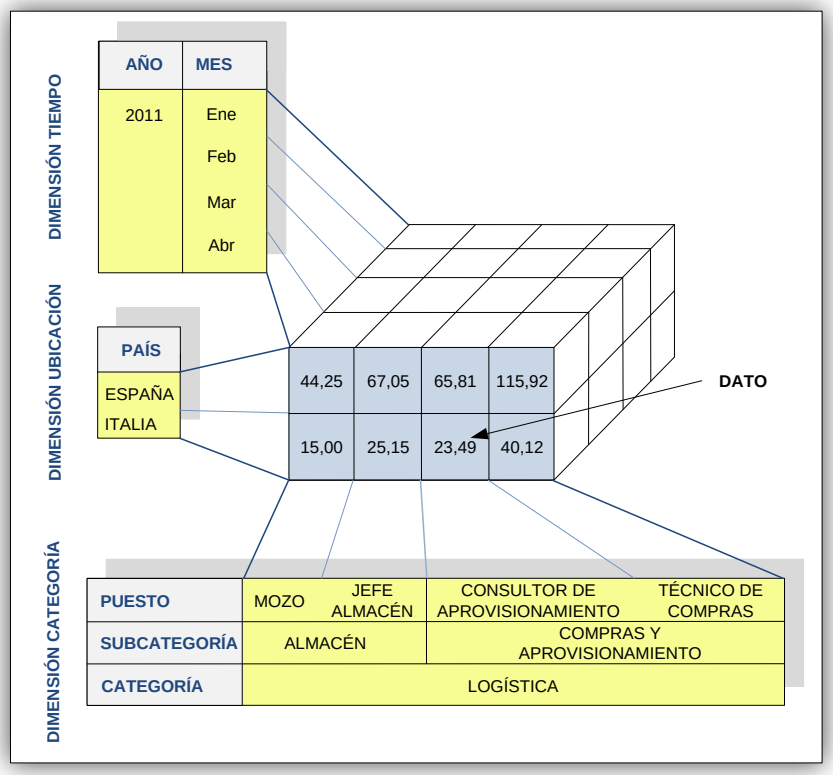


Ilustración 5. Ejemplo gráfico Datamart modificado

(6) Facilitar al usuario final la explotación de los *Datamart*

Una vez se ha homogeneizado la información y a concluido la *ETL* los datos deben ponerse a disposición del usuario, quien extraerá la información y conocimiento de estos para tomar las decisiones más acertadas.

Existen distintos caminos para analizar la información, esta fase se desarrolla en el punto siguiente.

2.3. Explotación de la información en un sistema de *BI*

La explotación de la información generada puede darse de diferentes formas. La *Explotación directa* y las *Tablas agregadas* son dos buenas opciones, pero en un sistema *BI* quizá los modelos unificados dimensionales sean la mejor opción, por lo que este documento se centrará en la explotación de *cubos multidimensionales*.

2.3.1. Explotación Directa

Mediante la ejecución directa de sentencias SQL contra el *SGDBR*, existen herramientas comerciales que encapsulan la generación y la ejecución de estas sentencias SQL de cara al usuario final que generalmente no es técnico. Se suelen utilizar cuando el usuario ataca a grandes *Datamart* con mucho volumen de información y permite la mayor flexibilidad a la hora de obtener informes.

2.3.2. Tablas Agregadas

Se trata de tablas con los datos más consultados por los usuarios. Cada una de estas tablas solventará una necesidad de información. Al realizar un pre-proceso de la información las consultas serán óptimas. Estas nuevas estructuras podrán ser consultadas desde aplicaciones web de desarrollo específico o desde las herramientas comerciales.

2.3.3. Cubos OLAP

Un paso más en la trasformación de datos. Se generan mediante herramientas comerciales que permiten recoger los datos y los guardan en estructuras internas. De cara al usuario, éste puede ver un cubo lógico y navegar por él. Las estructuras internas mejoran el rendimiento general de la navegación, y es la opción ideal para la explotación de pequeños *Datamarts* donde toda la información es de igual importancia.

La Ilustración 4. Ejemplo gráfico Datamart de este capítulo exponía un ejemplo de una parte de un cubo OLAP.

Se trata de un modelo dimensional, dónde los *hechos* y *dimensiones* de las que se componen los *cubos* han sido establecidas y relacionadas en el nivel de BD. Así, un *cubo* puede usar una, varias o todas las *dimensiones* creadas. Al mismo tiempo, estas *dimensiones* pueden definir otros *cubos*. Además, un *cubo* puede utilizar la misma *dimensión* varias veces.

Las *dimensiones* son un componente indispensable en los *cubos*, pues éstas organizan los datos en función del área de interés, como clientes, recursos… Las *dimensiones* contienen atributos que se corresponden con las columnas de las tablas de dimensiones. Estos atributos aparecen jerarquizados. Las jerarquías pueden definirse por el usuario del *cubo* y son útiles para organizar las *medidas*.

Una *dimensión de base de datos* es independiente de un *cubo*, y la instancia de una *dimensión de base de datos* en un cubo se denomina *dimensión de cubo*.

La *dimensión de base de datos* depende de la estructura de tabla o tablas que forman los *Datamarts*.

2.4. Productos de un sistema de *BI*

Los análisis realizados con las herramientas de explotación de información, descritas en el punto 2.3.Explotación de la información en un sistema de *BI*, son un producto de un sistema de *BI*. De los diferentes estudios se puede extraer toda la información contenida en el *DWH*.

Sin embargo, existen otros productos, entre los que cabe destacar los *Cuadros de Mando* (*CM*), *Sistemas de Soporte a la Decisión* (*DSS*) y *Sistemas de información Ejecutiva* (*EIS*).

2.4.1. Cuadros de Mando

Los *CM* están orientados a la alta dirección, son una herramienta pensada para establecer y supervisar los objetivos de una empresa.

Una vez los directivos han establecido el modelo de negocio que debe seguir la empresa en su totalidad, se utiliza dicho modelo para establecer unos *indicadores* (que no tienen porqué ser todos *indicadores* o *medidas* de de un *cubo*, si bien se basan en estos). Los *CM* tienen como fin realizar un seguimiento de estos *indicadores* definidos y no un detalle minuciosos de los datos.

El hecho de establecer un modelo de negocio global y traducirlo a *indicadores*, permite que toda la empresa sepa cómo alcanzar el modelo y permite hacer un seguimiento de la situación.

Pero los *Balanced Scoreboard* o *Dashboard*, como también se conoce a los *CM*, presentan una serie de riesgos. Dado que están basados en modelos de negocio, resultan una herramienta contraproducente en empresas con un plan estratégico en evolución, por otro lado, un modelo incompleto o poco trabajado no aporta valor y supone un trabajo en vano, además, los indicadores deben transmitir el mensaje por lo que deben escogerse meticulosamente y por supuesto, el *CM* no tiene sentido si está desfasado.

2.4.2. Sistemas de Soporte a la Decisión

Un *DSS* (en inglés, *Decision Support System*) es una herramienta enfocada al análisis de los datos de una organización a través de informes, que permiten resolver gran parte de las limitaciones de los programas de gestión.

Los informes que esta herramienta genera no requieren de un perfil técnico, se trata de informes dinámicos y flexibles, en una interfaz gráfica amigable y vistosa.

Los *DSS* están orientados a cualquier nivel gerencial dentro de una organización, en contraposición a los *CM* que se enfocan a la alta dirección.

La base de datos subyacente es un *DWH* o uno o varios *Datamarts* lo que aporta una serie de ventajas: el acceso a la información no requiere de grandes tiempos, se puede acceder a toda la información integrada de la organización (si bien esto no implica que todo usuario tenga a su disposición cualquier dato, estas herramientas permiten cribar la información según el perfil del usuario que accede a ellas) y se dispone de un histórico de la información, útil para realizar comparaciones entre periodos.

Turban, en 1995, [Turban E., 1995] definió un *DSS* como "*un sistema de información basado en un computador interactivo, flexible y adaptable, especialmente desarrollado para apoyar la solución de un problema de gestión no estructurado para mejorar la toma de decisiones. Utiliza datos, proporciona una interfaz amigable y permite la toma de decisiones en el propio análisis de la situación*".

Existen varios tipo de *DSS*, más específicos y orientados a un colectivo más concreto.

- **Sistemas de Información Gerencial (MIS)** (en inglés, *Management Information System*) sistema que analiza la información suministrada por el personal de la organización con el fin de ser útil a la administración y toma de decisiones.

- **Sistemas de información ejecutiva (EIS)** (en inglés, *Executive Information System*) son el tipo de *DSS* que más se suele emplear. Enfocada a gerentes, les permite acceder cómodamente a información interna y externa de su compañía, y que es relevante para el éxito de la organización.

- **Sistemas expertos basados en inteligencia artificial (ES)** (en inglés, *Expert System*) también se refieren a ellos como sistemas basados en conocimiento, utilizan redes neuronales para simular el conocimiento de un experto y utilizarlo de forma efectiva para resolver un problema definido.

- **Sistemas de apoyo a decisiones de grupo (GDSS)** (en inglés, *Group Decision Support Systems*) teniendo por base que mejorando las comunicaciones se mejoran las decisiones tomadas por un grupo de personas, se trata de un sistema que ofrece un entorno único de trabajo a un grupo de personas con un objetivo común.

2.4.3. Sistemas de información Ejecutiva

En ingles, *Executive Information System*. Es una herramienta software, basada en un *DSS*, que pone a disposición de los gerentes un panorama completo del estado de los indicadores de negocio. Cuentan con un acceso sencillo e intuitivo a información interna y externa de su organización, a fin de tener monitorizada la totalidad o una unidad de la organización y poder tomar decisiones que conduzcan al éxito a tiempo

Los informes generados por estos software suelen incluir alertas e informes basados en excepción, así como históricos y análisis de tendencias.

Al ser un tipo de *DSS*, se nutren de la información de un *Datamart* o un *DWH* y al presentar *indicadores* de negocio pueden asemejarse, en cierto modo, a un *CM*.

2.5. Conversión entre modelos

La construcción de un sistema de *BI* pasa por convertir los esquemas de datos existentes en esquemas multidimensionales, como esquemas en estrella o copo de nieve.

En el entorno de de las BD Relacionales el modelo E/R constituye una importante técnica de diseño de sistemas transaccionales. Hace posible la normalización de la estructura de datos física, obteniéndose un diseño sin redundancias en lo datos y de almacenamiento optimizado. Sin embargo, el modelado dimensional siendo mucho menos riguroso en cuanto a organización, permite a analistas y diseñadores más flexibilidad en el diseño, para lograr un mayor desempeño y optimizar la recuperación de la información, contribuyendo a facilitar las consultas del usuario sobre la BD resultante.

El modelo multidimensional produce una base de datos sobre la que es más sencillo realizar consultas, ya que hay menos cantidad de tablas y relaciones que en el modelo E/R. Esta reducción en el número de tablas relaciones y des-normalización, en muchos casos, de la información hace que no resulte fácil transformar un modelo E/R en un

modelo multidimensional aún cuando los datos que se modelan son los mismos. Ambos enfoques tienen diferentes puntos de partida, emplean técnicas distintas y producen resultados de diseño de BD diferentes. [Cedeño A., 2005]

Aplicaciones que precisan operar con datos heterogéneos y procedentes de diversas fuentes hacen necesaria la conversión entre modelos. Siendo un modelo una estructura compleja que representa un objeto de diseño e implicando esta conversión, en algunos casos, transformaciones en los datos del modelo [Castro Galán E., 2003].

Se pueden analizar tres aspectos generales en la conversión de modelos:

- **Integración de la información**: en las organizaciones y empresas la información reside en diferentes entornos operacionales, aunarla en un solo posibilita al usuario hacer uso de toda la información sin necesidad de cambiar la fuente.

- **Migración de datos**: Para realizar la migración de los datos debe existir un conocimiento externo de los mismos que haga posible la integración de todos en un esquema nuevo. No obstante, las diferencias entre los dominios, de donde proviene la información y donde se integra, hace que exista, en muchos casos, una pérdida de información.

- **Mezcla de ontologías**: Los diferentes entornos operacionales combinan diversas ontologías. Se debe hallar una que pueda suplir, mínimamente, a todas. Para establecer las equivalencias entre éstas se pueden emplear métodos de conversión de modelos que automaticen agilicen el proceso.

Además de estos aspectos se debe tener en cuenta que la representación explicita de la semántica hace que la equivalencia entre modelos no sea perfecta y automática en todos los casos, y se pueda realizar de forma individualizada en cada caso, a fin de conseguir completar el modelo resultante para evitar la pérdida de información que a veces se produce. Como consecuencia del uso de múltiples lenguajes de representación, debe permitirse la heterogeneidad, pudiéndose necesitar un nuevo lenguaje capaz de unificar [Castro Galán E., 2003].

Siguiendo a J. Madhavan [Madhavan, et. al., 2002] hay tres propiedades que toda equivalencia debe cumplir:

- **Habilidad para responder cualquier consulta del modelo:** el conjunto de consultas que se puede realizar sobre el modelo inicial debe poderse responder igualmente en el modelo resultante de la conversión.

- **Inferencia de fórmulas sobre la correspondencia:** las fórmulas que formalizan los modelos iniciales deben tener formulación equivalente en el modelo resultante que las aúna.

- **Composición de correspondencias:** dadas dos conversiones entre dos modelos donde uno es común en ambas, existe una correspondencia directa entre los otros dos modelos diferentes (de cada una de las conversiones primeras).

En la realización de este proyecto no se llega a la formulación matemática de los esquemas implementados, por lo que no puede aplicarse la segunda propiedad. Igualmente no se estudia la tercera propiedad, ya que sólo se realiza la conversión entre dos esquemas (*E/R* y *estrella*)

Es importante distinguir la conversión entre modelos de la conversión entre esquemas. La primera de ellas comprende un nivel más abstracto de la situación y por tanto más complejo, la segunda aborda casos con datos concretos. Existen investigaciones que abordan las conversiones entre modelos proponiendo herramientas, que basándose en las definiciones formales de los modelos, automaticen las conversiones [Bernstein, A., Levy, A. & Pottinger, R., 2000] y [Madhavan, et. al., 2002].

Capítulo 3

Este capítulo lo conforma la descripción de las herramientas utilizadas para el desarrollo de los objetivos de este proyecto final de carrera. IBM Cognos 8 como herramienta de procesamiento analítico de datos e Informática Power Center como instrumento para la transformación entre modelos o esquemas de datos. Además se hará referencia a SQL Server 2000 como sistema de origen de la información.

Se referirán los componentes de cada herramienta y sus funcionalidades. Se analizarán alternativas a estos programas y por último se elaborará una planificación temporal y presupuesto del proyecto.

3. ESTUDIO DE LA VIABILIDAD Y GESTIÓN DEL PROYECTO

Un objetivo principal que se persigue con este estudio es la conversión de un esquema *E/R* en un *esquema en estrella* y en un *esquema en copo de nieve*.

Otro objetivo, también importante, es el estudio de la información contenida en el *esquema en estrella*, mediante un *cubo multidimensional*.

El esquema de datos *E/R*, implementado en una base de datos SQL Server 2000, será completado con información ficticia a través de un proceso *ETL* manual, haciendo uso de herramientas, globalmente conocidas, como Microsoft Office Excel.

La conversión de este esquema inicial *E/R*, en uno *en estrella* y en otro *en copo de nieve*, en la práctica diaria se realiza con herramientas específicas para la transformación entre modelos con opciones de optimización y mejora del rendimiento de las máquinas que lo ejecutan. Por esto, se analizará una herramienta comercial, Informática Power Center 9.1.

Sin embargo, se limitará al estudio general de esta herramienta concreta, ya que el coste de este tipo de programas supera al presupuesto del proyecto, en donde se realizará la conversión entre modelos a través de la creación de vistas en SQL Server, aunque las consultas en lenguaje SQL carezcan del potencial de Informática PowerCenter en muchos aspectos.

3.1. Herramienta Power Center. Componentes y funcionalidades

Con esta herramienta se puede realizar una *ETL*. PowerCenter, abreviado en adelante como PWC, presenta un interfaz amigable e intuitiva. Diferentes objeto identifican las operaciones que se pueden desarrollar y representa el flujo de datos entre transformaciones a través de flechas que conectan los objetos.

El proceso *ETL*, extracción, transformación y carga de los datos, es parte del ciclo de vida de una implementación de *BI*.

Las fases de este proceso general son tres:

- **Extracción**: Procesos de extracción de la información de los sistemas orígenes. Para ello se debe evaluar los datos que este sistema contiene.

- **Transformación**: Procesos de transformación de la información para adecuarla al modelo final.

- **Carga**: Procesos de carga en la base de datos destino, adaptada al modelo que se adecua a las necesidades concretas.

PWC está compuesto por cuatro herramientas cliente:

- Repository Manager (R)
- Designer (D)
- Workflow Manager (W)
- Workflow Monitor (M)

Además el Servicio de Repositorio y el Servicio de Integración establecen las conexiones con los orígenes de información y ejecutan los procesos.

Servicio del repositorio (Repository Service)

Administra conexiones al Repositorio de PWC desde los clientes del Repositorio. Un cliente del repositorio es cualquier componente PWC que conecta al repositorio.

El Servicio del Repositorio es un proceso multi-hilo que inserta, actualiza, obtiene metadatos en las tablas del repositorio y asegura la consistencia de los datos que hay en él.

Servicio de Integración (Integration Service)

El Servicio de Integración conecta con el repositorio para obtener los metadatos almacenados y ejecuta las tareas de los *workflows*.

Un *workflow* es un conjunto de tareas que describen cómo y cuándo ejecutar trabajos relacionados con la extracción, transformación y carga de los datos. Una *sesión* es un tipo de tarea de un *workflow*, que describe cómo mover los datos desde las fuentes a los destinos usando un *mapeo* o *mapping*.

Una *sesión* extrae datos desde las fuentes de los *mapeos* y almacena los datos en memoria mientras les aplica las transformaciones configuradas en el *mapeo*, para posteriormente cargar los datos transformados en los destinos del *mapeo*.

La arquitectura global PWC es la siguiente:

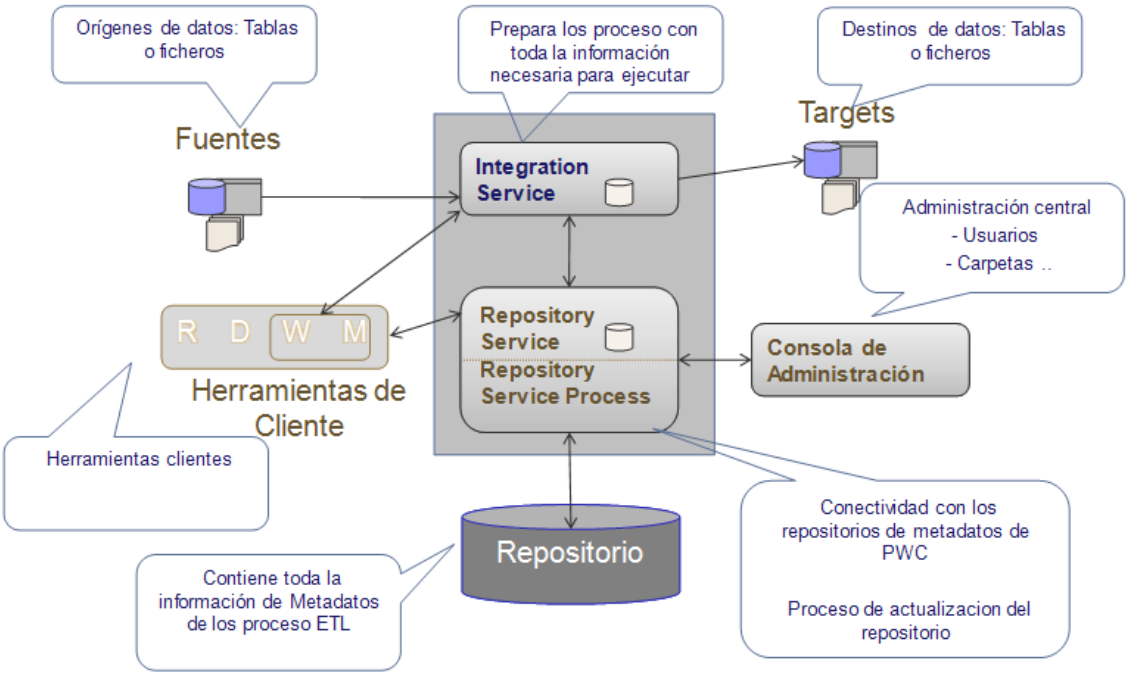


Ilustración 6. Arquitectura global PWC

3.1.1. Repository Manager

Componente que permite gestionar usuarios y administrar los elementos creados.

Los usuarios se definen en función de las necesidades de cada uno, pudiendo ser administradores y desarrolladores.

Los elementos creados se organizan en directorios, el administrador debe crearlos a los desarrolladores, especificando las propiedades del directorio.

Esta herramienta se utiliza para llevar los elementos desde un entorno (directorios) de desarrollo al entorno de explotación.

Además con Repository Manager se puede realizar búsquedas sobre los elementos desarrollados y analizar las dependencias existentes entre ellos.

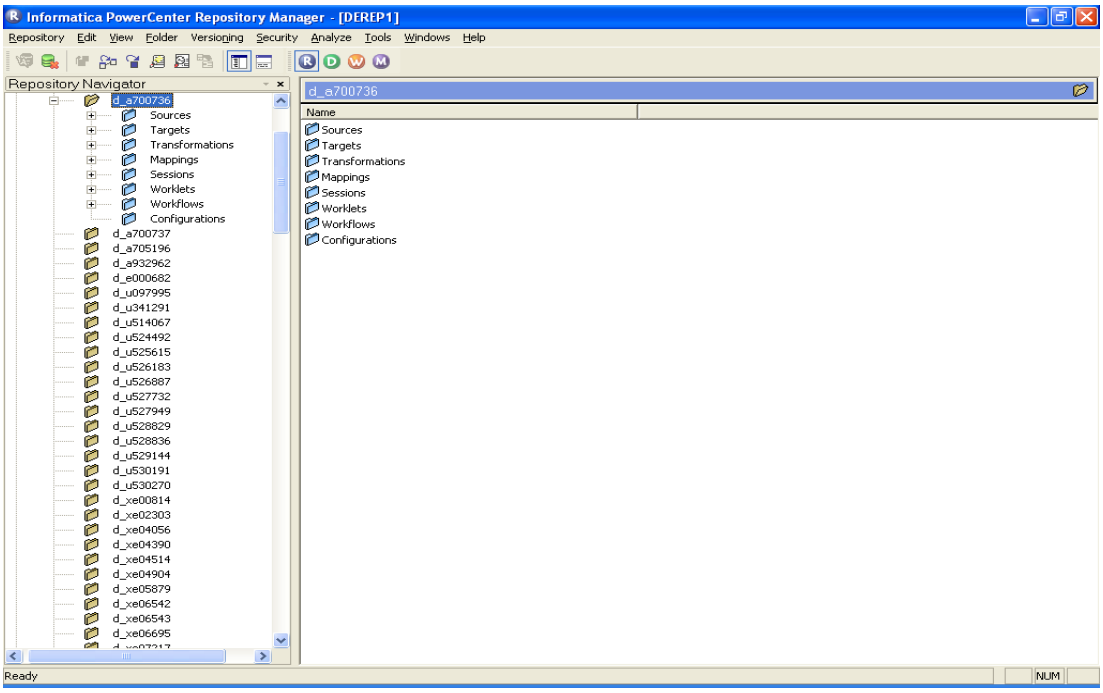


Ilustración 7. Ventana principal Repository Manager

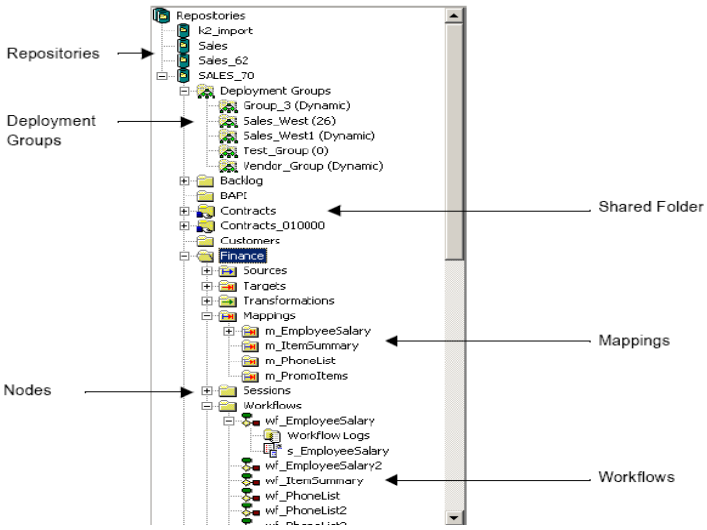


Ilustración 8. Directorios y elementos Repository Manager

Funcionalidades del interfaz

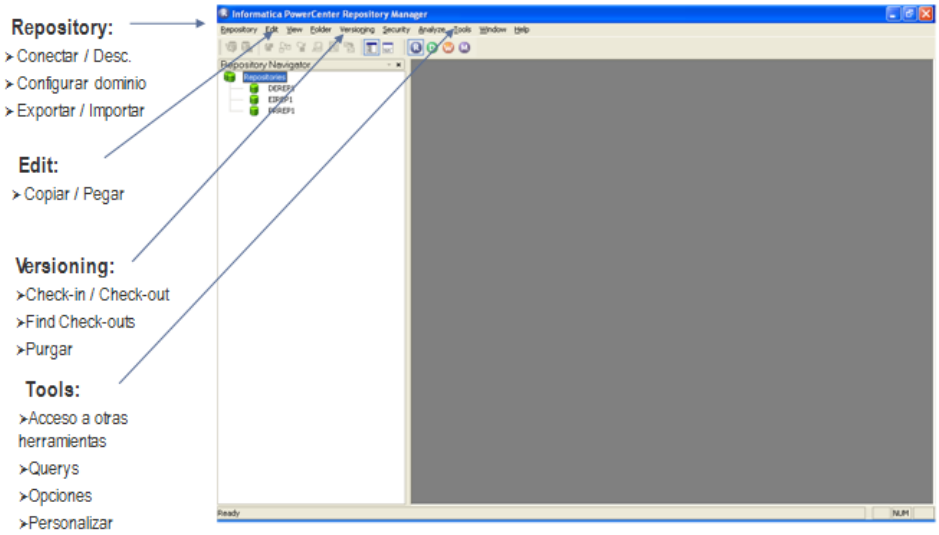


Ilustración 9. Funcionalidades interfaz Repository Manager

3.1.2. Designer

Herramienta donde se desarrollan los *mapeos* que describen el conjunto de operaciones efectuadas sobre los datos para transformarlos. En los *mapeos* existen básicamente tres puntos clave: el origen de los datos (los *sources*), el tratamiento que reciben (transformaciones que les aplican) y lugar de almacenamiento final (los *target*).

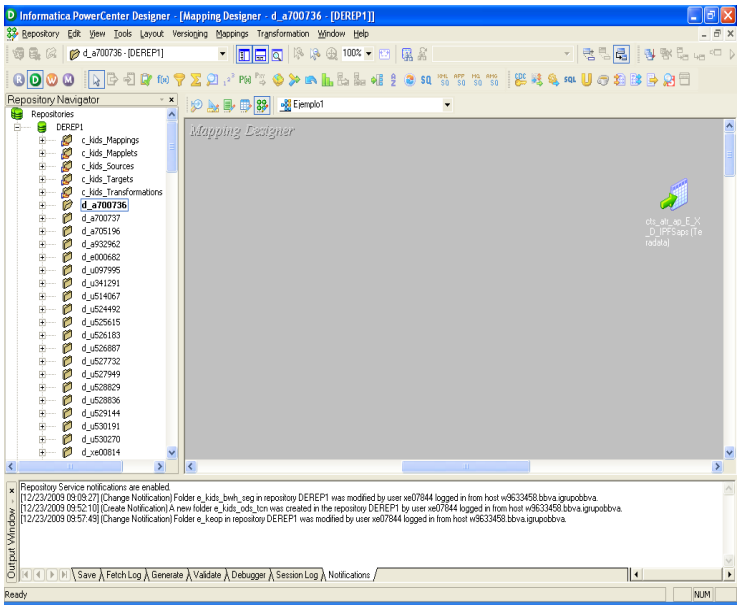


Ilustración 10. Ventana principal Designer

Funcionalidades de la interfaz

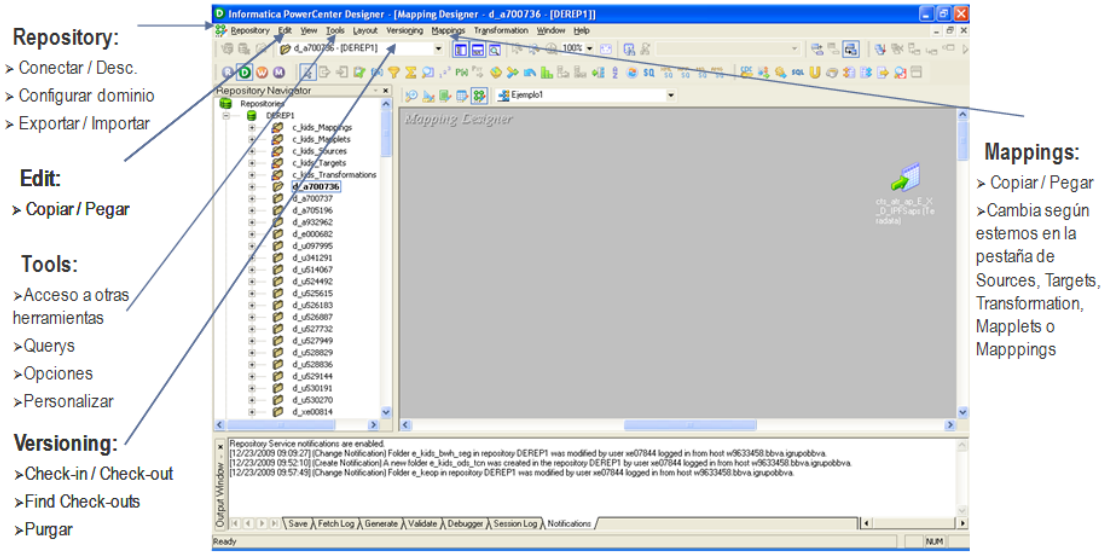


Ilustración 11. Funcionalidades interfaz Designer

Un *mapeo* es un conjunto de transformaciones organizadas en un flujo que tienen uno o más orígenes de datos y almacenan la información transformada en uno o varios destinos.

Para la realización de un *mapeo* se deben establecer los orígenes de datos, cuyos campos son arrastrados desde los puertos de salida a los puertos de entrada de la siguiente transformación del flujo hasta llegar a las tablas de destino.

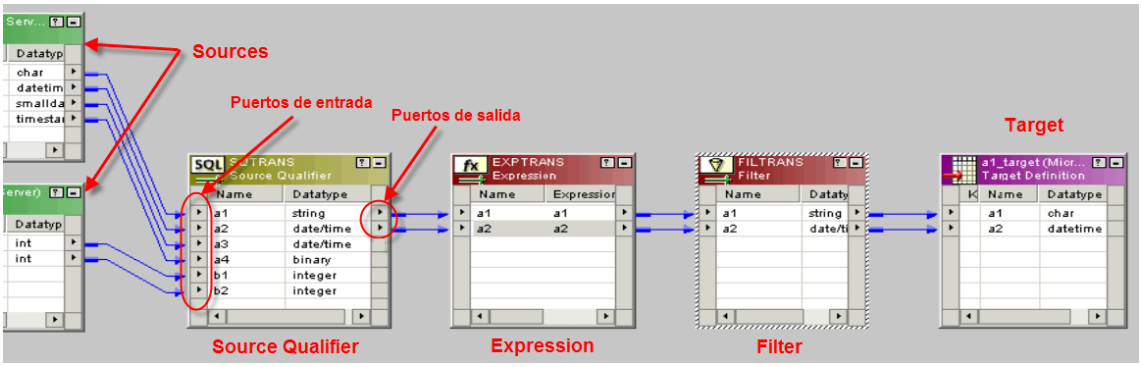


Ilustración 12. Flujo dentro de un mapeo

3.1.3. Workflow Manager

Con este componente se gestionan los mapeos desarrollados. Aquí se crean *sesiones* (una o más por cada *mapeo*) y se establece el flujo entre ellas, además nos permiten establecer la periodicidad con que las referidas *sesiones* se ejecutarán.

El flujo de ejecución de las *sesiones* forma un *workflow*. Este flujo puede ser de sesiones secuenciales o concurrentes.

Ilustración 13. Flujo de sesiones en serie

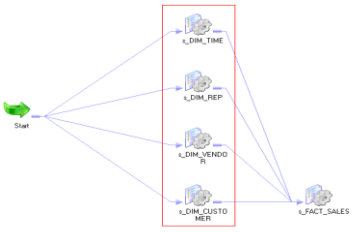


Ilustración 14. Flujo de sesiones en paralelo

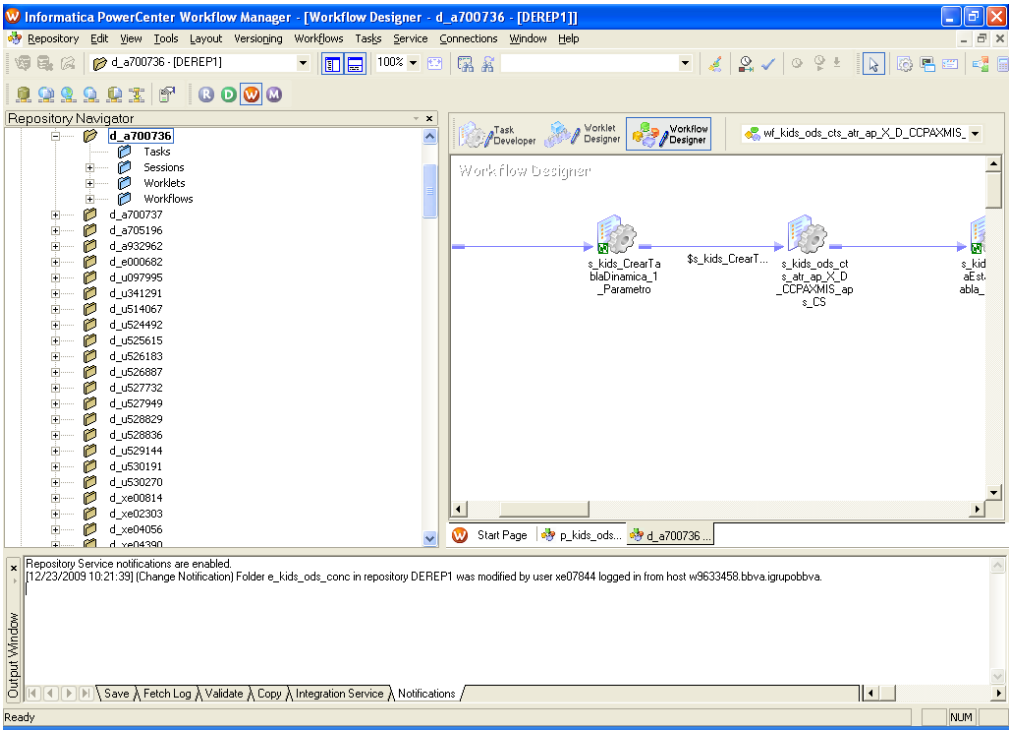


Ilustración 15. Ventana principal Workflow Manager

Funcionalidades de la interfaz

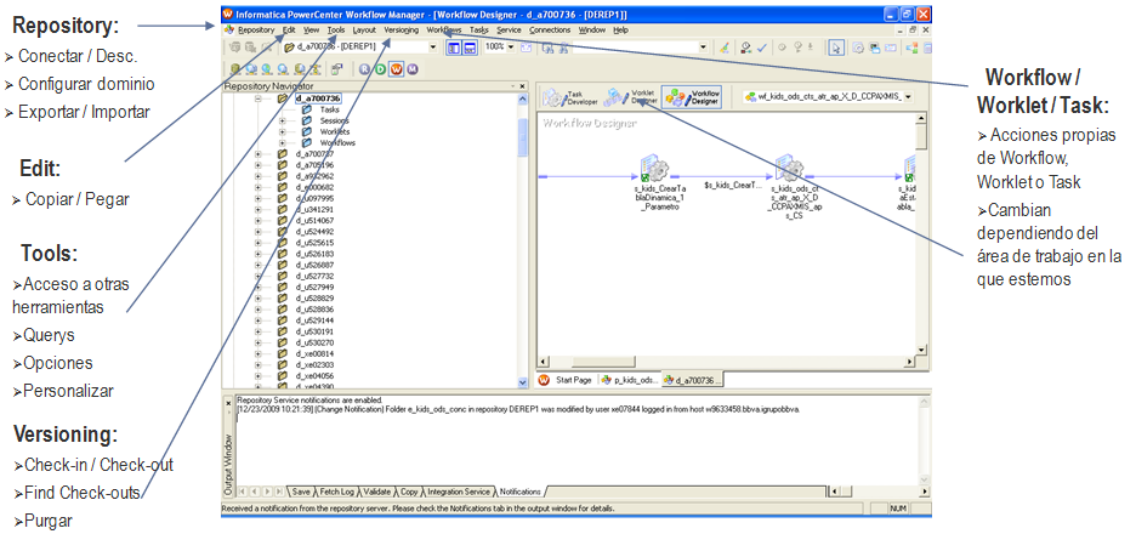


Ilustración 16. Funcionalidades interfaz Workflow Manager

3.1.4. Workflow Monitor

Para realizar el seguimiento a la ejecución de los procesos, se dispone de esta herramienta. En la ventana se observan los diferentes estados por los que van pasando los procesos, el detalle de la ejecución y los errores se producen, además desde la aplicación se puede acceder a los archivos de log generados por las diferentes ejecuciones.

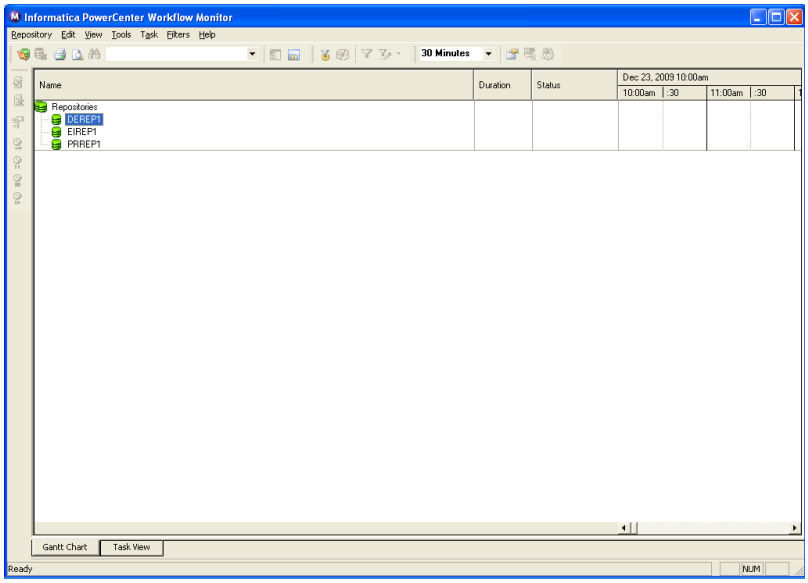


Ilustración 17. Ventana principal Workflow Monitor

Funcionalidades de la interfaz

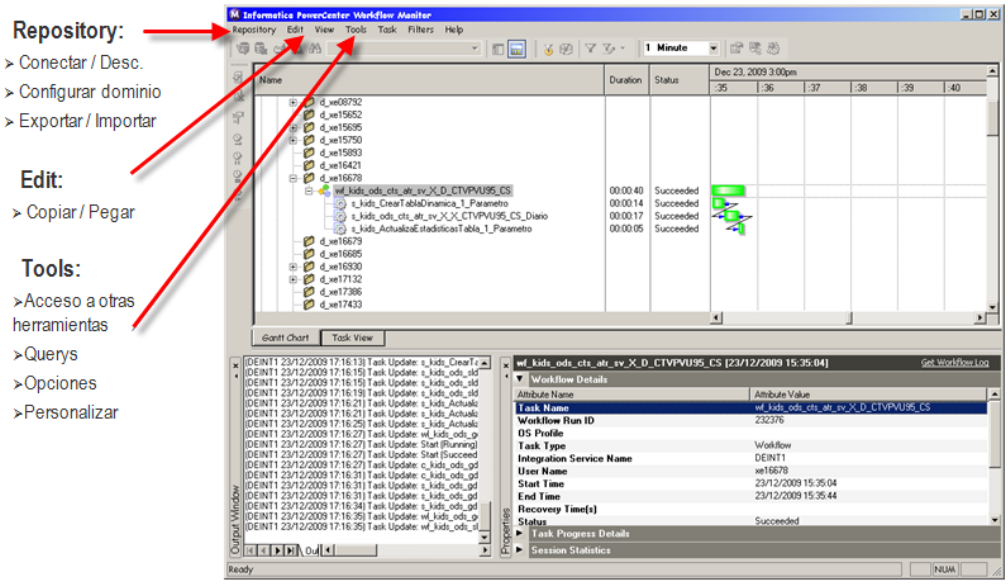


Ilustración 18. Funcionalidades interfaz Workflow Monitor

3.2. Herramienta Power Center. Transformaciones básicas.

En el diseño de los *mapeos* se pueden emplear una serie de transformaciones de los datos predefinidas por la herramienta. El uso de una u otra transformación dependerá del diseño que haga el desarrollador.

Cada transformación tiene *puertos* que corresponden a los diferentes campos de las tablas. Estos *puertos* pueden pasar de una transformación a otra seleccionando y arrastrando. Los puertos pueden ser de entrada (los datos que entran), salida (datos que salen) o variable (calculados)

Las características, y recomendaciones de uso, de estas transformaciones se detallan en este punto.

3.2.1. Source Qualifier (Origen de datos)

Extrae los registros desde ficheros planos o tablas relacionales y los convierte a tipos de datos propios de PWC.

Recomendaciones para el óptimo rendimiento de la transformación

Con el fin de reducir el volumen de datos a tratar, utilizar las propiedades del Source Qualifier para seleccionar únicamente aquellos registros que sean necesarios en el proceso, y emplear solo los puertos necesarios

Si por el contrario se mantienen todos los puertos, independientemente de si son utilizados o no, los cambios futuros serán menos costosos.

La consulta asociada al Source Qualifier puede modificarse manualmente alterando el código SQL que genera. Esta opción no es recomendada, ya que si se hace el resto de propiedades de la transformación, que pretenden optimizar el proceso, serán ignoradas.

3.2.2. Expression (Función)

Permite la definición de puertos de tipo variable. Este tipo de puertos son equivalentes a declaraciones de variables locales a la transformación que permiten simplificar expresiones complejas, optimizando los cálculos.

Pueden aplicarse un gran número de funciones sobre los datos, agrupadas en la herramienta son las siguientes:

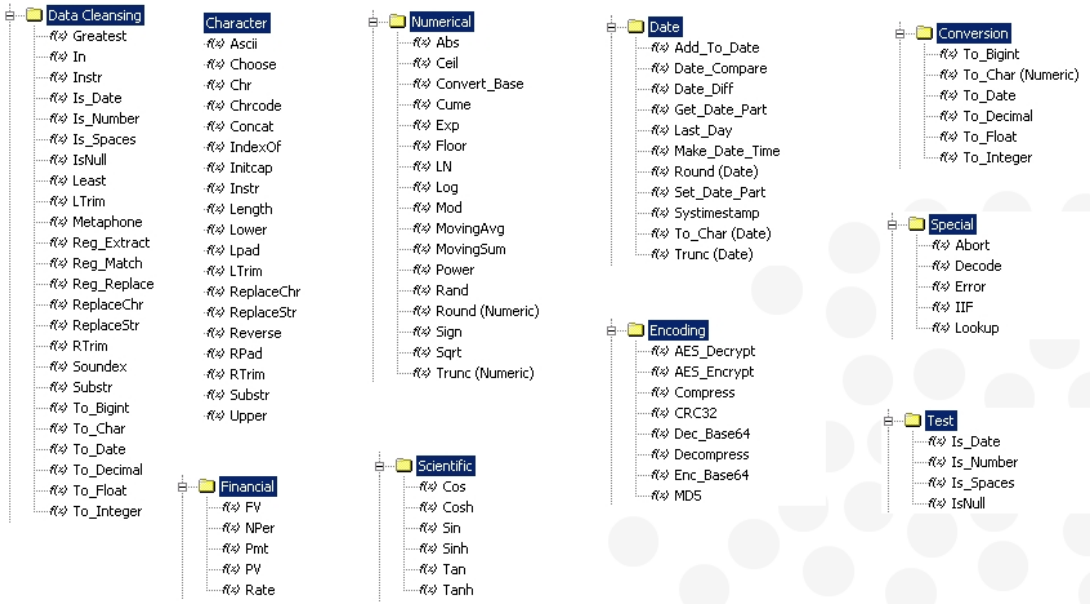


Ilustración 19. Funciones predefinidas en transformaciones

Recomendaciones para el óptimo rendimiento de la transformación

Con el fin de reducir el volumen de datos a tratar, incluir únicamente aquellos *puertos* que sean necesarios para los cálculos requeridos.

Cada transformación de este tipo procesa por separado los puertos incluidos en la misma. Si es necesario realizar varios cálculos sobre un mismo conjunto de *puertos* utilizar una única transformación Expression. De este modo se procesan los datos una única vez.

3.2.3. Filter (Filtro)

Es utilizada para descartar registros del flujo de datos en base a una o más condiciones que son evaluadas para cada registro.

Cualquier expresión que devuelva un único valor puede ser usada como filtro (0 equivale a falso y cualquier otro valor es verdadero). Además están disponibles los operados lógicos AND y OR y valores constantes.

Los campos con posibles valores nulos también son susceptibles de ser filtrados, con funciones condicionales combinadas con la expresión IsNull.

Recomendaciones para el óptimo rendimiento de la transformación

Filtrar lo antes posible en un flujo de datos reduce el volumen de datos a tratar.

Siempre que sea posible utilizar las propiedades del Source Qualifier para seleccionar los registros necesarios, en lugar de utilizar la transformación Filter.

Evitar expresiones complejas en la condición de filtrado. Si es necesario, se puede calcular el resultado de evaluar la condición de filtrado con una transformación Expression, un campo de salida, que funcione como marca, muestra si se filtra o no el registro.

Para los filtros que pueden incluirse en el resto de transformaciones se recomienda igualmente las expresiones complejas, dando la misma alternativa de usar una transformación Expression previamente.

3.2.4. Sorter (Ordenación)

La transformación de tipo Sorter produce una ordenación de los datos de entrada en función de los *puertos* especificados (uno o más). Este tipo de transformaciones solo tiene *puertos* de entrada y salida, no se pueden realizar cálculos en ellas.

Recomendaciones para el óptimo rendimiento de la transformación

Parametrizar, en la medida de lo posible, suficiente memoria en el parámetro de caché de la transformación para evitar que se vuelque dicho caché a disco.

3.2.5. Aggregator (Agrupación)

Esta transformación realiza cálculos agregados sobre agrupaciones de datos. Los datos por los que se agrupa la información se definen sobre los *puertos* de entrada.

Es el único tipo de transformación en el que se permiten operaciones de agregación. Los operadores permitidos son: *avg* (media aritmética), *count* (cuenta), *first* (primero de un conjunto), *last* (último del conjunto), *max* (máximo), *min* (mínimo), *percentile* (Percentil), *stddev* (Desviación estándar), *sum* (suma) y *variance* (varianza). Estos operadores pueden anidarse, por ejemplo, *max* (*count* (variable)).

Recomendaciones para el óptimo rendimiento de la transformación

Es mucho más óptimo agregar por campos de tipo numérico que de tipo cadena o fecha.

La entrada de datos ordenados reduce el volumen de datos a mantener en los cachés de la transformación, si llegan ordenados a la transformación, se debe marcar la opción que lo indica, Sorted Input, para que los datos sean gestionados como ordenados.

Limitar el número de *puertos* de tipo entrada/salida para reducir el volumen de datos que la transformación ha de mantener en caché.

Utilizar *puertos* de tipo variable, para optimizar los cálculos donde una misma expresión haya de evaluarse varias veces para un mismo registro.

3.2.6. Joiner (Cruce)

Se utiliza para cruzar fuentes de datos de localizaciones heterogéneas.

Para cada cruce se debe especificar el tipo de unión, que origen de datos que funcionara como maestro de la información (*Master*) y el cuál será el detalle (*Detail*) y los *puertos* que funcionarán como clave de cruce.

Los tipos de unión definidos son:

- **Normal Join.** Basándose en la condición de cruce definida, el servicio de integración descarta todos los registros de las fuentes *Master* y *Detail* que no tengan ninguna coincidencia,

- **Master Outer.** Mantiene todos los registros de la fuente *Detail*, y del *Master*, solamente los registros que tengan alguna coincidencia en el cruce. En los registros *Detail* que no coincidan con ninguno del *Master*, los valores de los campos correspondientes al *Master* se establecerán a NULL.

- **Detail Outer.** Inverso del Master Outer, mantiene todos los registros de la fuente *Master*, y del *Detail*, solamente los registros que tengan alguna coincidencia en el cruce. En los registros *Master* que no coincidan con ninguno del *Detail*, los valores de los campos correspondientes al *Detail* se establecerán a NULL.

- **Full Outer.** Se obtienen todos los registros de ambas fuentes. Para los registros de una de las fuentes que no encuentren alguna coincidencia en la otra fuente, los valores de los campos de la fuente que falte, serán establecidos a NULL.

Recomendaciones para el óptimo rendimiento de la transformación

Utilizar, siempre que sea posible, la entrada de datos ordenados, si no es posible definir como *Master* el flujo de datos con menor número de registros minimiza los recursos requeridos por la transformación. Si se utiliza la propiedad de entrada de datos ordenada, definir como *Master* el flujo de datos con menor número de valores duplicados para la clave de cruce reduce al mínimo el uso de cachés.

Los Normal Join son más rápidos que los Outer Join, ya que reducen el volumen de datos a tratar.

Si se requiere cruzar más de dos flujos de datos con transformaciones Joiner en cascada, cruzar los flujos de menor a mayor en cuanto a número de registros.

3.2.7. Lookup (Consulta)

Realiza búsquedas de los datos de entrada en otras tablas o ficheros planos especificados, basándose en la clave establecida para la búsqueda.

Recomendaciones para el óptimo rendimiento de la transformación

Resulta más eficiente utilizar un Joiner en lugar de un Lookup para cruzar flujos de datos con volúmenes muy elevados.

Mantener todos los *puertos* en la transformación Lookup, independientemente de si son utilizados o no, no impacta en el rendimiento y facilita futuros cambios en el proceso.

3.2.8. Update Strategy (Estrategia de carga)

Este tipo de transformación marca los datos según sea la estrategia de carga: insertar, actualizar, borrar o rechazar.

Recomendaciones para el óptimo rendimiento de la transformación

Considerar estrategias alternativas a la captura de errores de validación de datos si es posible que se detecte un número elevado de los mismos. Estos registros rechazados se incluyen en el log de ejecución de la sesión y opcionalmente en el fichero de registros rechazados impactando directamente en el rendimiento del proceso.

Separar las operaciones de inserción, actualización y borrado sobre un mismo destino para facilitar su monitorización.

Utilizar la estrategia de carga incremental en lugar de la opción Update else Insert a nivel de sesión para asegurar el óptimo rendimiento de la misma.

3.2.9. Router (División de la información)

Divide los datos en varios grupos, que se usan generalmente en flujos diferentes, en base a múltiples condiciones definidas, en las que pueden utilizar funciones y operadores lógicos.

Los grupos no son excluyentes entre sí a menos que las condiciones asociadas a los mismos lo sean. Un mismo registro puede pertenecer a más de un grupo. Y los registros que no cumplan ninguna de las condiciones pertenecen al grupo Default.

Recomendaciones para el óptimo rendimiento de la transformación

Utilizar una transformación Router en lugar de múltiples transformaciones Filter, ya que cada transformación Filter procesa el flujo de datos completo, mientras que la Router procesa los datos una única vez para evaluar todas las condiciones, mejorando los tiempos de ejecución.

Es importante de cara al mantenimiento del proceso, utilizar un nombre descriptivo para cada grupo definido en la transformación Router.

Asociar todos los registros del flujo a un grupo y no usar el grupo Default mejora los tiempos de ejecución al reducir la complejidad de la consulta SQL (*query*).

3.2.10. Unión (Unión)

Reúne los datos de diferentes flujos. Pueden existir tantos orígenes como se desee, pero todos deben tener la misma estructura inicial y final. La transformación Union no elimina aquellos registros duplicados que puedan resultar una vez combinados los grupos de entrada en un mismo flujo de datos.

Recomendaciones para el óptimo rendimiento de la transformación

Utilizar un nombre descriptivo para cada grupo de entrada permite determinar a simple vista la relación entre los mismos.

La transformación Union puede utilizarse prácticamente en cualquier punto del flujo de datos.

3.3. Herramienta Cognos 8. Componentes y funcionalidades

Este proyecto aborda la implementación de un *cubo multidimensional* basado en el *esquema en estrella*, producto de la conversión del esquema *E/R*. Para esta implementación se utiliza la herramienta comercial de IBM, Cognos 8.

Este programa es una solución de *BI* basada en Web, con funciones integradas de creación de informes, análisis, tablas de puntuación y administración de eventos. Incluye una serie de componentes que se pueden dividir entre los orientados al servidor y los orientados de modelado.

Para la realización del *cubo* y su análisis no es preciso el uso de todos los componentes disponibles. Este punto del documento realizará una descripción de estos componentes y se ahondará en aquellos de interés para el proyecto.

3.3.1. Componentes de servidor

Los siguientes componentes proporcionan la función del servidor para el enrutamiento y el procesamiento de las solicitudes del usuario, además de las interfaces de usuario para la creación de análisis, informes, tablas de puntuación y administración de eventos, así como.

3.3.1.1. IBM Cognos Configuration

Componente necesario para establecer la configuración inicial del resto de módulos de Cognos 8, una vez han sido instalados, o para modificar los parámetro de estos

posteriormente. Se utiliza igualmente para iniciar o detener el servicio de un componente en el equipo local.

IBM Cognos Configuration puede ejecutarse en modo interactivo o bien en modo silencioso. En modo interactivo, se utiliza una interfaz gráfica de usuario para configurar el componente de Cognos. En modo silencioso, la herramienta se ejecuta en segundo plano y, como parte de una instalación desatendida, no requiere la intervención del usuario. De forma predeterminada, funciona en modo interactivo.

Después de especificar los valores para las propiedades de configuración, es recomendable utilizar la función de prueba disponible para validar los valores de las propiedades. Esta validación puede probar los nodos de la configuración por separado o todos al mismo tiempo.

Si los cambios efectuados no son válidos o han dejado de ser adecuados para el entorno, existe la opción de restablecer los valores predeterminados, bien de todas las propiedades de un grupo o componente, o bien de una sola.

Para poder utilizar los componentes instalados, debe iniciar el servicio en todos los equipos donde instaló el servidor o los componentes de gestión de recursos, operación que puede requerir de unos minutos.

En Windows, el servicio se inicia automáticamente al iniciar el equipo. En UNIX, el proceso se inicia y se reinicia, pero debe configurarlo para que se ejecute como daemon.

La conexión de la BD correspondientes a la base de datos del almacén de contenido, al que se hará referencia en este mismo capítulo, es imprescindible para poder iniciar el servicio.

Del mismo modo que se debe iniciar el servicio para operar con él es posible detenerlo o reiniciarlo, bien un componente concreto, seleccionándolo, o todo el conjunto.

Es posible salir de IBM Cognos Configuration en cualquier momento. Dejando el servicio en el mismo estado, ejecutando o parado.

Interfaz de usuario de IBM Cognos Configuration

A través de la interfaz gráfica de usuario se configuran los valores de las propiedades, como los números de puerto, las ubicaciones de los datos y los tamaños de archivo.

Estos ajustes se almacenan en la carpeta de configuración del equipo en el que estén instalados.

La interfaz de usuario de IBM Cognos Configuration consta de:

Ventana Explorador

La ventana Explorador muestra una lista de grupos organizados en áreas funcionales. Estos grupos pueden contener componentes (sólo si están instalados) y propiedades.

Seleccionando un componente que contenga propiedades, se puede cambiar los valores en la ventana propiedades.

La ventana Explorador puede contener la totalidad o una parte de los iconos siguientes:

Icono	Descripción
	Configuración en el equipo local.
	Un grupo, que es un conjunto de propiedades relacionadas, organizado en áreas funcionales, de los componentes instalados en el equipo local.
	Un componente de IBM Cognos que puede contener propiedades y uno o varios recursos.
	Un recurso que representa un archivo externo, una base de datos u otro objeto utilizado por el componente de IBM Cognos. Un recurso puede contener otros recursos.

Ventana Propiedades

En ella se muestran los valores de las propiedades del componente o recurso seleccionado en la ventana Explorador. En esta ventana se puede cambiar o restablecer el valor predeterminado de una propiedad.

La ventana Propiedades puede contener la totalidad o una parte de los siguientes iconos.

Icono	Descripción
	El valor de la propiedad es necesario para que se ejecute la configuración local.
	El valor especificado no es el predeterminado.
	Haga clic en el icono para escribir valores en un cuadro de diálogo.
	El valor especificado está fuera de los límites requeridos o es un tipo de dato incorrecto, o bien falta un valor obligatorio.

Ventana Información

Muestra ayuda sensible al contexto para el grupo, el componente, la propiedad o el recurso seleccionado en un determinado momento. También muestra los posibles mensajes de error relacionados con el valor de una propiedad seleccionada.

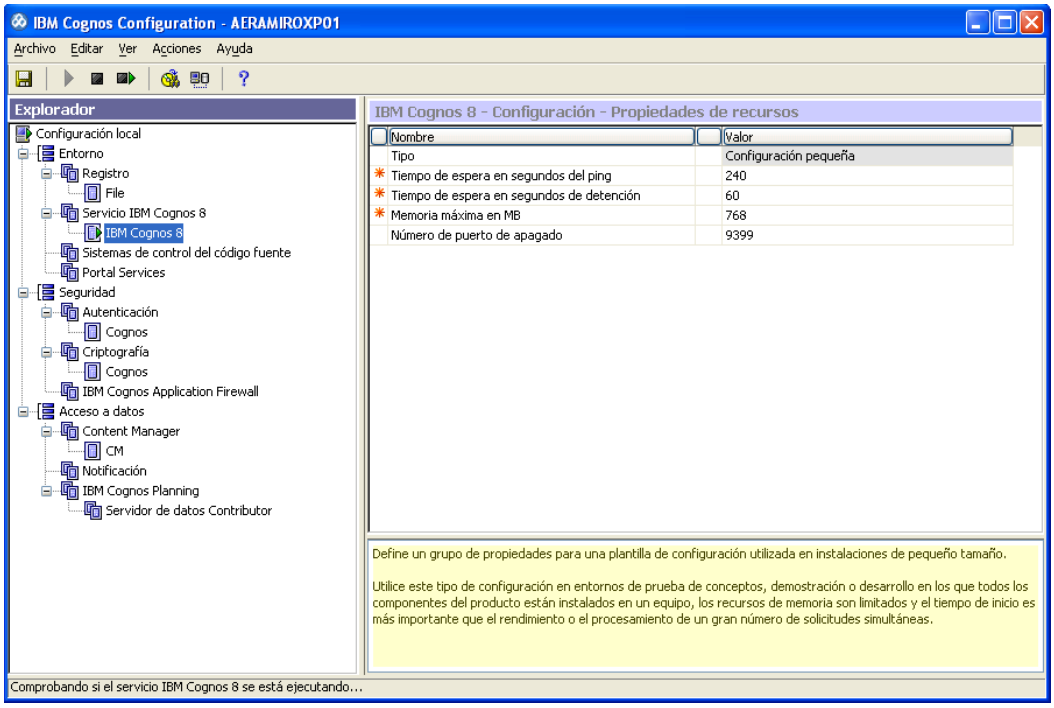


Ilustración 20. Cognos Configuration

Opciones de la línea de comandos

Una alternativa a la interfaz gráfica, es la línea de comandos con el comando de configuración para modificar el comportamiento de IBM Cognos Configuration cuando se inicia.

3.3.1.2. Content Manager

Content Manager es el servicio que administra el almacenamiento de los datos de aplicaciones del cliente, la seguridad (cuando procede), los datos de configuración, los modelos, los indicadores, las especificaciones de informes y las salidas de informes.

El servicio es necesario para publicar paquetes, recuperar o almacenar especificaciones de informes, administrar información de planificación y administrar el espacio de nombres de Cognos.

Este componente almacena toda esta información automáticamente y de forma transparente al usuario, en una base de datos relacional, que constituye el Almacén de Contenido, otro tipo de componente que se detalla más adelante, en este punto y cuya conexión es imprescindible para poder iniciar el servicio en Cognos.

El almacén de contenido es un recurso del Content Manager. La definición de un recurso contiene propiedades que describen la ubicación del recurso o la manera en que el componente de IBM Cognos lo utilizará.

Para algunos componentes se debe tener como mínimo un recurso. De lo contrario, no podrá guardarse la configuración, es el caso del Content manager.

3.3.1.3. IBM Cognos Connection

Todas las aplicaciones de IBM Cognos 8 basadas en Web se pueden ejecutar a través de IBM Cognos Connection.

Se trata de un portal Web que proporciona un único punto de entrada para realizar consultas, analizar y organizar los datos, así como para crear informes, tablas de puntuación y eventos.

Además este componente puede integrar otras aplicaciones de Business Intelligence y direcciones URL a otras aplicaciones.

A través de este componente se accede a los servicios web que ofrece la herramienta, se realiza a través del servidor donde. En este caso, Cognos se encuentra instalado en local, por lo que la ruta de acceso en el navegador es: http://localhost/cognos8/

El acceso a la web puede tener una seguridad asociada y dependiendo del perfil del usuario que acceda y los componentes instalados, podrá visualizar una serie de servicios o herramientas.

En el proyecto la web está alojada en un servidor local sin seguridad asociada, por lo que el usuario no precisa de autenticación y accede con el rol de Administrador.

Ilustración 21. Página inicio IBM Cognos Connection

Esta página inicial se muestra en la primera conexión al servicio y es opcional, puede continuar mostrándose en cada inicio del servicio o bien mostrarse en su lugar la página definida como principal. En cualquier caso, todas las acciones expuestas en esta página inicial pueden realizarse a través del menú 'Iniciar' presente en la web una vez se inicia Cognos.

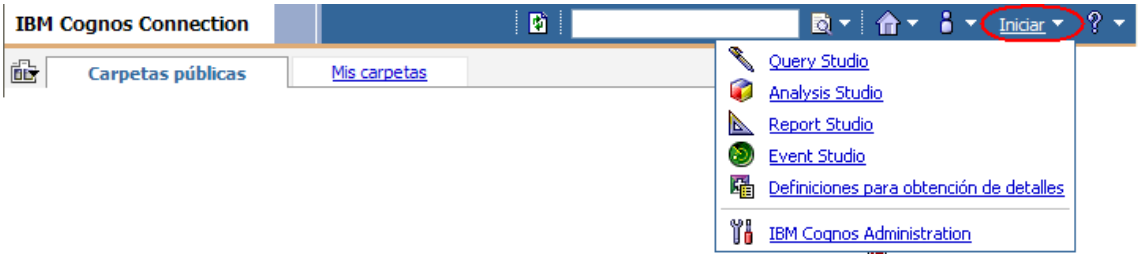


Ilustración 22. Opciones menú Iniciar

Para un usuario administrador, la página de inicio de Cognos agrupa las herramientas y opciones en tres menús:

- **Mi contenido**: Agrupa las opciones que permiten llevar al usuario al contenido almacenado en Cognos, modelos, cubos, informes, vistas predefinidas…

- **Administración**: opción para gestionar el portal. Permite administrar las actividades que se están ejecutando o están planificadas (ejecución de informes, de Cuadros de Mando…), la seguridad el portal y las opciones de configuración del portal (selección de un estilo general de visualización del portal).

- **Mis acciones**: Muestra las acciones principales que pueden realizarse. Este menú puede variar en función de los componentes de IBM Cognos instalados.

A continuación se profundiza en las diferentes opciones mostradas en los menús.

Mi contenido

Mi página principal

La opción 'Mi página principal' sitúa al usuario en la página principal del portal, por defecto es la página 'Carpetas públicas', ya definida por la aplicación. Esta página puede cambiarse y será la que muestre en caso de decidir omitir la página de inicio (anteriormente descrita).

Contenido de IBM Cognos

La opción 'Contenido de IBM Cognos' constituye una serie de páginas representadas como pestañas en la parte superior de la web. Por defecto se dispone de dos, 'Carpetas públicas' y 'Mis Carpetas', estas pestañas muestran el sistema de directorios que el usuario crea en la herramienta, los modelos y cubos publicados (carpetas azules), los diferentes informes y análisis de cubos realizados que han sido guardados (vistas predefinidas guardadas). Los modelos pueden contener carpetas, como un directorio más.

El formato de las páginas 'Carpetas públicas' y 'Mis Carpetas' esta predefinido y solo es posible cambiar el estilo por otro ya creado.

Las páginas creadas por el usuario pueden personalizarse, mostrando no solo un sistema de directorios. Es posible incluir imágenes, texto HTML embebido, accesos directos a direcciones web, informes en formato pdf… Las páginas creadas pueden definirse como página principal.

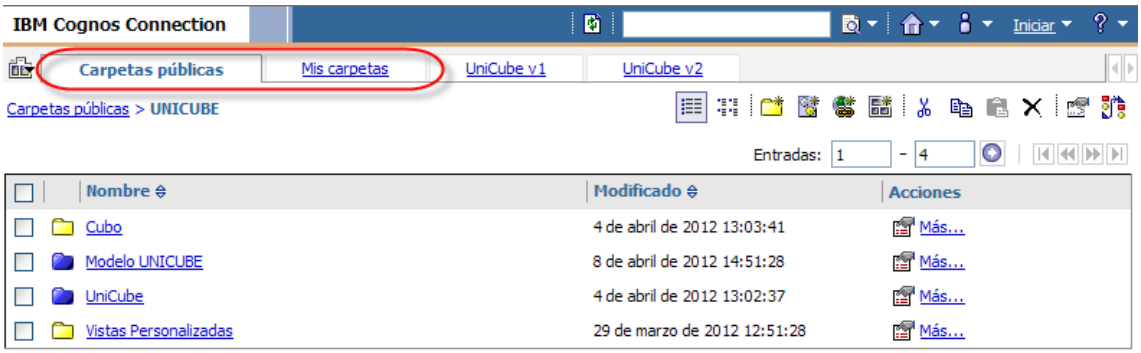


Ilustración 23. Pestañas o vistas predefinidas en Cognos Connection

Este es un ejemplo de página personalizada con imágenes y menús horizontales: que no muestra un sistema de carpetas:

Ilustración 24. Pestañas o vistas personalizadas en Cognos Connection

Administración

Administración Contenido de IBM Cognos

Seleccionando la esta opción se accede a la administración del portal de Cognos 8 desde la que es posible llevar un seguimiento de los procesos pasados, actuales y planificados en Cognos, establecer grupos de usuarios y directivas de seguridad para ellos y configurar los paquetes de datos y las conexiones con las bases de Datos y paquetes de información generados con otras herramientas de modelado de Cognos 8, no integradas en el portal, como Framework Manager o Transformer con las que se modela paquetes de datos y cubos multidimensionales respectivamente.

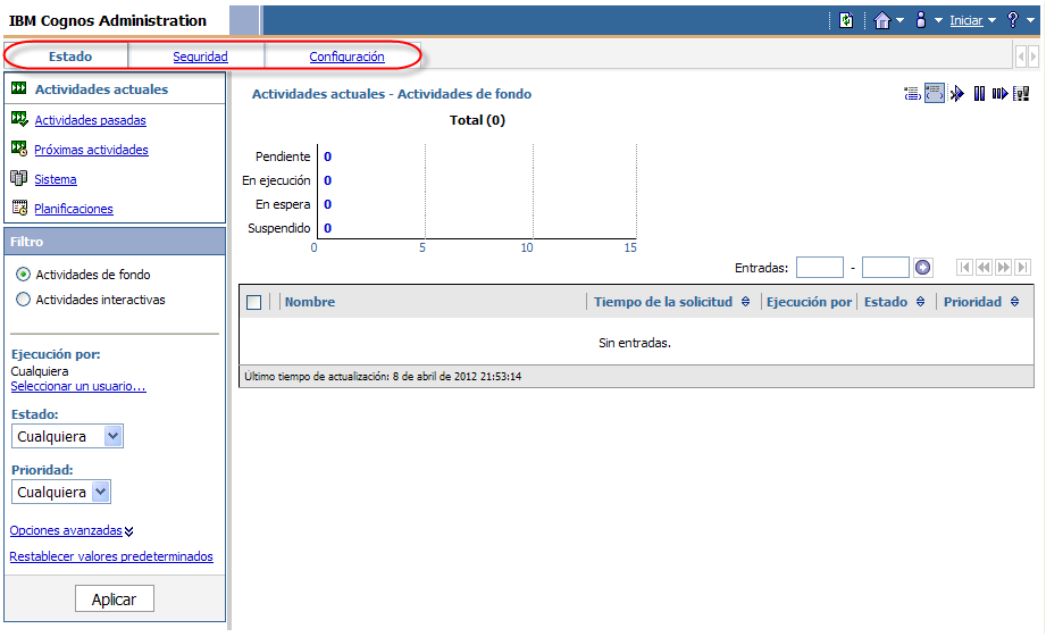


Ilustración 25. Ventana de administración en Cognos Connection

Mis acciones

En este menú aparecen opciones que ejecutan componentes de Cognos 8. Con la instalación básica de Cognos 8 se establecen elementos por defecto, algunos de los cuales no serán estudiados en detalle en este proyecto. Todos los componentes son descritos a lo largo del punto.

Consultar mis datos

La opción 'Consultar mis datos' ofrece al usuario la opción de analizar la información publicada con el componente Query Studio.

Tras pulsar la opción aparece un menú con los últimos paquetes analizados (con esta herramienta o con otros componentes de Cognos) y la opción de navegar por los directorios del portal para seleccionar nuevos paquetes.

Ilustración 26. Ventana de selección de paquetes

Analizar mi empresa

La opción 'Analizar mi empresa' ofrece al usuario la opción de analizar la información publicada con el componente Analysis Studio, este es un componente adicional, con instalación independiente a la de Cognos 8.

Seleccionando esta opción se accede a un menú donde seleccionar el paquete de datos a analizar, análogo al mostrado con la opción 'Consultar mis datos'. Una vez indicado el paquete de datos se muestra la herramienta.

Crear informes profesionales

Con la opción 'Crear informes profesionales' se accede a la herramienta Report Studio, instalado por defecto con Cognos 8.

Al igual que con las dos opciones anteriores, es preciso seleccionar el paquete de datos con el que se realizará el informe. Una vez elegido se accede a Report Studio.

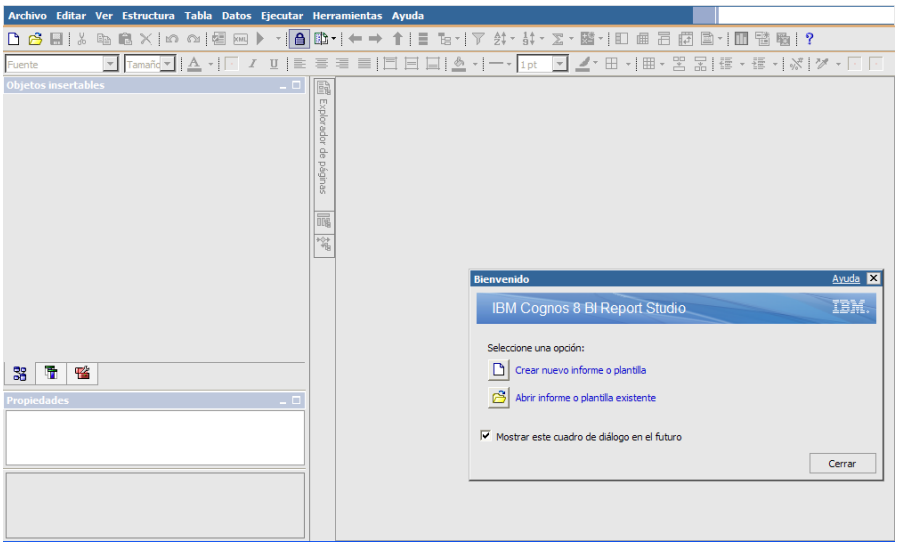


Ilustración 27. Ventana inicial Report Studio

Administrar mis eventos

A través de la opción 'Analizar mis eventos' el usuario accede al componente, instalado por defecto con Cognos 8, Event Studio, con el que se pueden crear y administrar los eventos.

Una vez seleccionada la opción y el paquete de datos sobre el que se realizará el evento se inicia la herramienta.

El estudio de este componente queda fuera del alcance del proyecto.

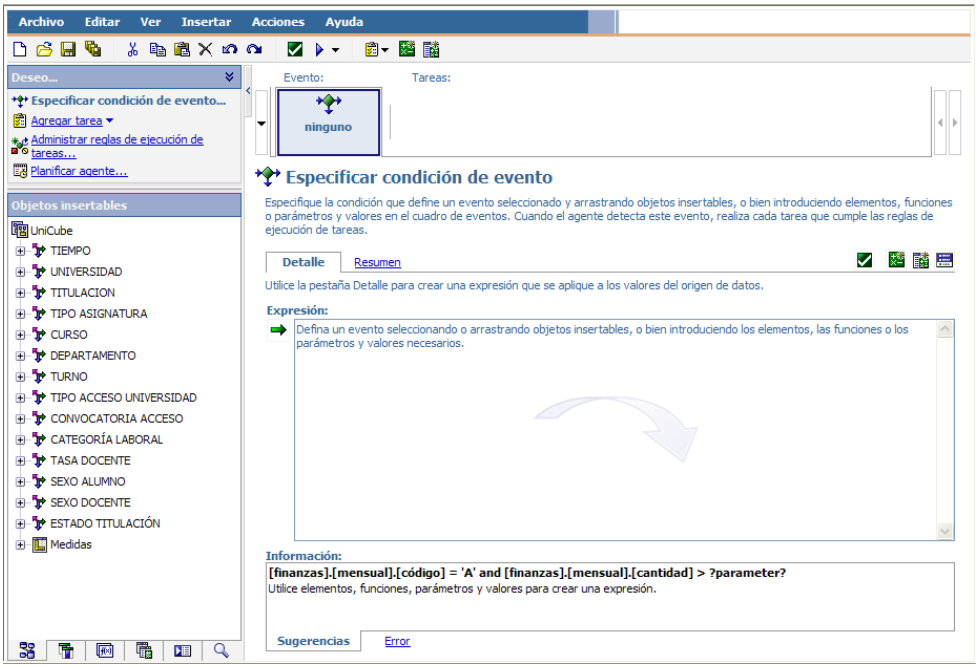


Ilustración 28. Ventana inicial Event Studio

3.3.1.4. Cognos Viewer

Es un portlet al que se puede acceder a través de IBM Cognos Connection y de cualquier portal empresarial existente. Permite ver e interactuar con cualquier tipo de contenido de IBM Cognos publicado.

3.3.1.5. Analysis Studio

Proporciona acceso a orígenes de datos dimensionales. Gracias a esta aplicación se puede explorar, analizar y comparar estos datos.

Analysis Studio permite examinar la información de un cubo multidimensional previamente diseñado y generado con la herramienta Transformer.

Los atributos de hechos y las dimensiones, definidos en el modelo publicado, pueden manejarse cómodamente en esta herramienta permitiendo análisis muy rápidos y potentes de los datos almacenados.

Los operadores dimensionales soportados por la herramienta, así como la navegación dimensional se definen en los siguientes puntos del capítulo.

En el ANEXO A - Manual de Usuario Analysis Studio se ofrece un manual de usuario para la herramienta con casos prácticos de ejemplo.

La interfaz de la herramienta puede dividirse en cuatro partes importantes:

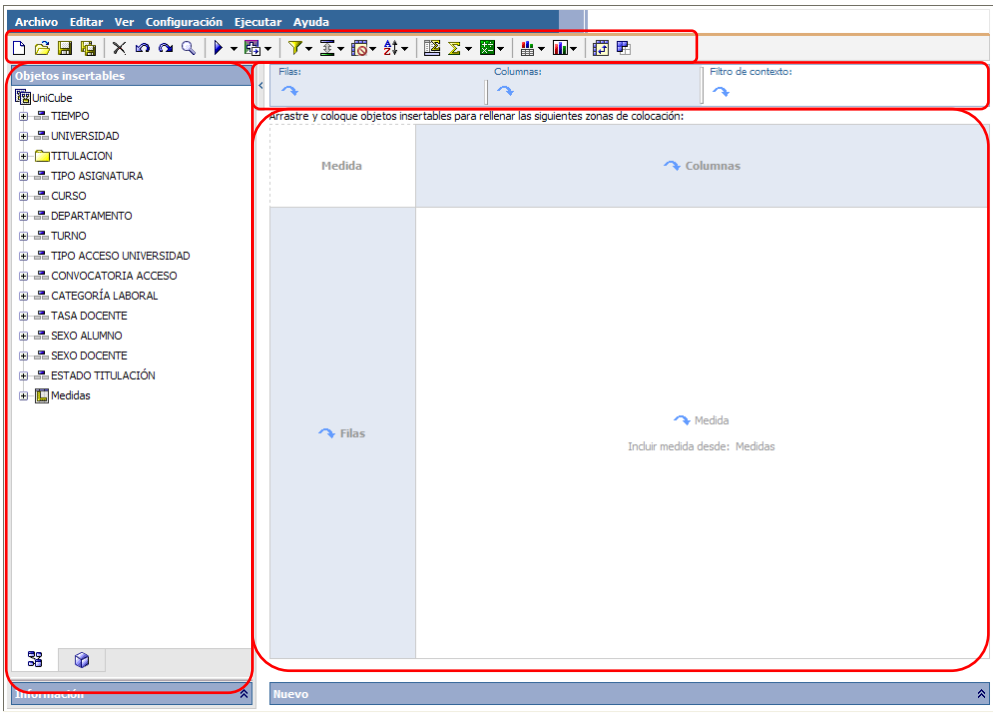


Ilustración 29. Análisis cubo UNICUBE

Objetos insertables

En el panel de la izquierda se hallan los **objetos insertables**. Son los elementos que es posible seleccionar para realizar el análisis deseado. Se distinguen Dimensiones y Medidas.

Espacio de trabajo

En el panel de la derecha se descubre el **espacio de trabajo**. En este espacio se arrastrarán los objetos insertables para realizar un análisis.

Panel superior

El **panel superior** del espacio de trabajo, en él se muestra los datos que se visualizan en las filas, las columnas y los filtros que se estén aplicando durante el análisis. Es importante no perder nunca de vista este panel ya que dota de significado a los datos analizados.

Por ejemplo, en la siguiente imagen se observa que el análisis muestra (aunque no se nos muestre en filas y columnas los datos están siendo filtrados por año académico).

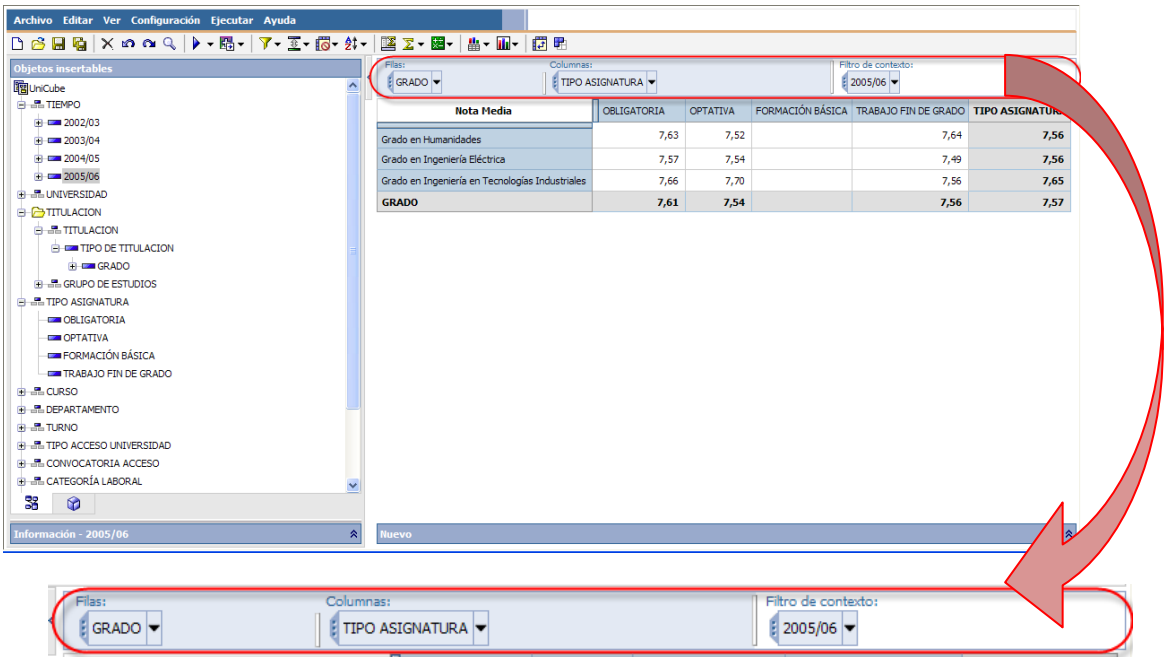


Ilustración 30. Panel superior

Barra de herramientas

En la parte superior se sitúa la **barra de herramientas**, donde se ofrecen las diferentes acciones que se pueden realizar.

Ilustración 31. Barra de herramientas

VISTA EN EL CUBO	DESCRIPCIÓN
Nuevo	Limpia el espacio de trabajo para realizar un nuevo análisis.
Abrir	Abre otra vista.
Guardar	Guarda una vista del espacio de trabajo.
Guardar como	Guarda una vista del espacio de trabajo con el nombre que se especifique.
Eliminar	Elimina un elemento en el análisis, por ejemplo una dimensión.
Deshacer	Deshace una acción. Útil si se comete un error.
Rehacer	Rehace una acción.
Buscar	Busca en los datos del cubo.
Ejecutar informe	Con esta opción se exporta la vista visualizada en pantalla múltiples formatos (entre ellos Excel y PDF)
Ir a	Se muestran enlaces vinculados al cubo.
Filtro	Aplica filtros a filas o columnas.
Mejores o peores	Ordena los valores en forma de ranking según una dimensión.
Suprimir	Suprime del análisis las filas y/o columnas que contengan nulos.
Ordenar	Ordena ascendente/descendentemente la fila o columna seleccionada.
Subtotal	Muestra el subtotal de las filas o columnas seleccionadas.
Resumir	Muestra una fila o columna con un nuevo resumen (suma, media, valor máximo, valor mínimo, varianza, etc.).
Calcular	Muestra un cálculo definido por el usuario (suma, resta, multiplicación, división, etc.) en el que intervienen varias filas o columnas.

VISTA EN EL CUBO	DESCRIPCIÓN
Mostrar	Muestra un gráfico con los datos mostrados en pantalla.
Tipo de gráfico	Cambia el tipo de gráfico a mostrar (columnas, barras, circular, líneas, etc.).
Cambiar filas y columnas	Cambia las filas por las columnas trasponiendo la tabla visualizada.
Guardar como conjunto personalizado	Crea un grupo personalizado de varios elementos para facilitar su manejo durante los análisis.

Otros componentes del servidor que quedan fuera del alcance del proyecto son:

3.3.1.6. Report Studio

Posibilita crear, editar y distribuir una amplia gama de informes profesionales. Asimismo, pueden definirse plantillas de informes estándar para utilizarlas en otros componentes como Query Studio.

Los análisis creados en Query Studio y Analysis Studio se pueden abrir y editar en Report Studio.

3.3.1.7. Query Studio

Es un componente ideado para usuarios con poca o ninguna formación específica de diseño. Permite crear y guardar rápidamente informes que satisfacen necesidades que no cubren los informes profesionales estándar (creados en Report Studio).

La herramienta Query Studio permite el análisis de la información de forma similar al componente Analysis Studio, analiza los datos presentándolos en filas y/o columnas, puede anidarlos y contiene otras opciones análogas a Analysis Studio pero la representación de la información es más basta si no se personaliza.

Además del análisis de los datos y las opciones para personalizar la representación de la información, esta herramienta permite generar modelos de informes a partir los análisis realizados enlazando con el componente Report Studio, permitiendo generar informes de forma más intuitiva y sin necesidad de un conocimiento extenso del modelo de datos.

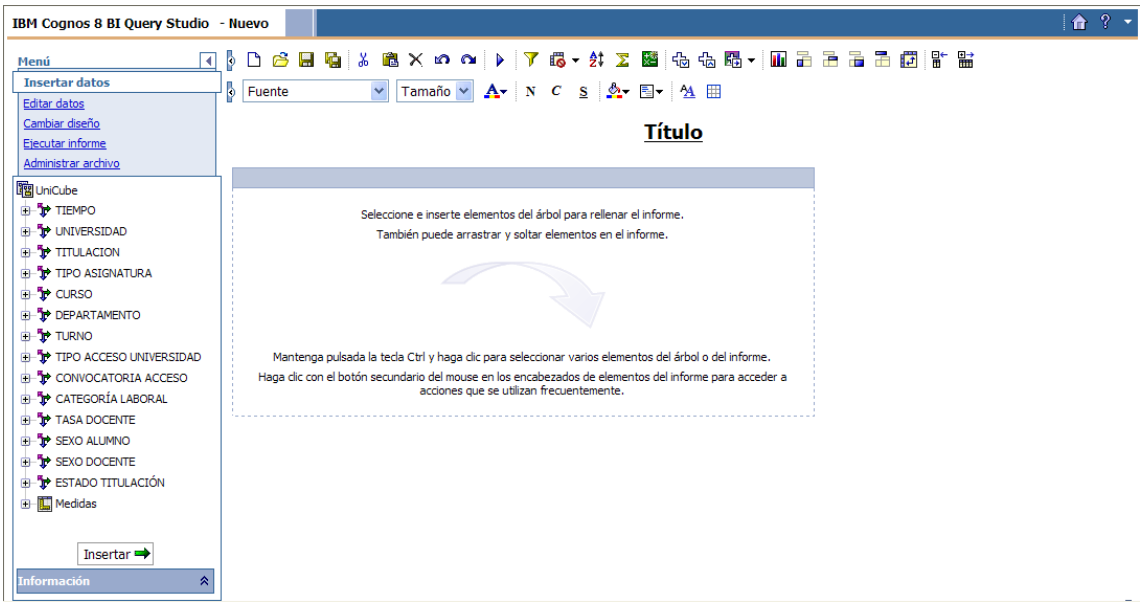


Ilustración 32. Ventana inicial Query Studio

3.3.1.8. Event Studio

Para configurar agentes que controlen los datos y realicen tareas cuando se dan condiciones excepcionales en los datos que deben tratarse.

Al producirse un evento, el usuario recibe una alerta que le informa.

Entre las acciones de los agentes se encuentran: publicar detalles en el portal, enviar alertas por correo electrónico, ejecutar y distribuir informes basados en eventos y controlar el estado de los eventos.

3.3.1.9. Metric Studio

Permite crear tablas de puntuación personalizadas, basadas en indicadores inter-funcionales, para controlar y analizar los indicadores en toda la empresa.

Los usuarios pueden controlar, analizar y notificar datos actualizados en las tablas de puntuación.

Son estos indicadores los que utilizan para crear Cuadros de Mando (2.4.1. Cuadros de Mando).

3.3.1.10. Puerta de enlace

La comunicación Web en IBM Cognos 8 se suele realizar a través de puertas de enlace, que residen en uno o varios servidores Web. Suelen ser programas CGI, pero pueden seguir otros estándares, como ISAPI (Internet Server Application Program Interface) y módulos de Apache.

3.3.1.11. Base de datos de contenido de Cognos

Utilizada para guardar las configuraciones internas de IBM Cognos 8, es una instancia de una base de datos Apache Derby. Esta instancia no puede utilizarse como base de datos de consultas.

Se trata de una alternativa si no desea implementar otra base de datos para el almacén de contenido.

3.3.1.12. IBM Cognos 8 Virtual View Manager

El componente proporciona acceso a orígenes de datos adicionales tales como LDAP, JDBC, Open XML y WSDL y mejora el rendimiento cuando se consultan datos de diferentes orígenes.

Es un componente muy práctico para las empresas, pues cuentan con LDAP corporativos, útiles en la implementación de seguridad en Cognos, y donde con frecuencia la información que se analiza se reparte en diferentes BB.DD. y servidores.

3.3.1.13. IBM Cognos 8 Go! Office

Establece un puente entre IBM Cognos 8 y Microsoft Office, permitiendo acceder a los datos de los productos de creación de informes de IBM Cognos en las aplicaciones Microsoft Office (se excluyen los informes que utilizan indicadores creados con Metric Studio)

3.3.2. Componentes de modelado

Como indica su nombre, estos componentes permiten el modelado de datos de los orígenes. Pudiéndolos estructurar y presentar de una forma significativa para el usuario final que analiza la información.

Los componentes de modelado incluyen las herramientas siguientes:

3.3.2.1. Framework Manager

Es la herramienta de modelado que crea y administra metadatos para así utilizarlos en el análisis y la creación de cubos e informes.

Los metadatos se publican en paquetes proporcionando una vista única e integrada de cualquier número de orígenes de datos heterogéneos.

Son estos paquetes publicados los que las herramientas de creación de cubos e informes utilizan.

La interfaz de la herramienta muestra una completa barra de herramientas que permite además de una visualización de innumerables elementos, la configuración de los parámetros necesarios para conectar con los orígenes de datos y publicar los paquetes con los modelos creados.

Las partes de la interfaz son:

Vista general del proyecto (Project Viewer)

En la parte izquierda se sitúa el mapa general del modelo Framework Manager abierto. Los diferentes orígenes de datos llamados Data Sources, las tablas creadas a partir de ellos, los espacios o tablespaces donde se organizan lógicamente las tablas, al conjunto de tablespaces se denomina Modelo o Model, el mapa de parámetros o Parameters Maps y los paquetes para publicaciones o Pakages.

Vista principal

En la parte central superior se visualiza la misma información que en la barra lateral izquierda pero separadamente y con un formato de iconos o lista según desee el usuario. Además el Modelo se completa en esta ventana mostrando las relaciones que se pueden establecer entre las tablas de los tablespaces (que no son visibles en la barra lateral izquierda).

Propiedades (Properties)

En la parte central inferior se observan las características y parámetros configurados de los elementos seleccionados.

Herramientas (Tools)

A la derecha se muestran las herramientas principales: resúmenes de los elementos creados en el proyecto, acciones posibles como verificación del modelo o publicación de los paquetes, un buscador de elementos o las dependencias establecidas en el proyecto o modelo.

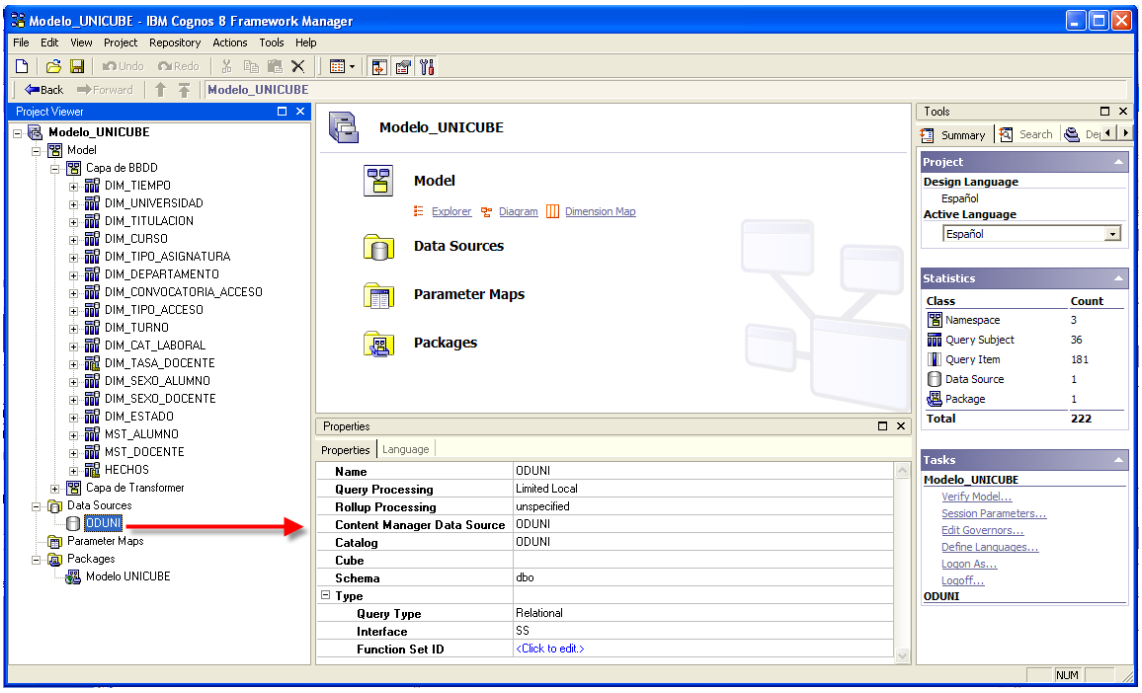


Ilustración 33. Ventana principal Framework Manager

3.3.2.2. IBM Cognos 8 Transformer

Es la herramienta de modelado que se emplea para crear cubos multidimensionales, utilizables por otros componentes de Cognos 8. Incorpora Power Play, una herramienta con una interfaz poco refinada que permite visualizar los cubos generados.

Es un componente muy potente que incorpora numerosos modelos de navegación y operadores multidimensionales. Capaz de leer de orígenes de datos dispares que van desde paquetes publicados a través del Framework manager a ficheros de texto plano con campos separados por comas, pasando por bases de datos Microsoft Access o hojas de cálculo Microsoft Excel.

La interfaz de Transformer muestra de manera predeterminada los principales elementos del modelo de un cubo en este herramienta, que son:

Mapa de dimensiones (Dimension Map)

Muestra un esquema general de las dimensiones creadas en el modelo, sus diferentes niveles, representado la jerarquía de los elementos de la dimensión, siempre ordenados descendentemente.

Desde esta vista se pueden configurar los elementos de una dimensión, el nombre, su origen de datos, el tipo de datos, las etiquetas que mostrarán, la ordenación dentro del nivel de la dimensión, etc.

En esta ventana se configuran opciones más avanzadas que permiten localizar cambios o errores en los orígenes de datos. Por ejemplo, cuando un elemento X de jerarquía inferior pertenece a un elemento A de la jerarquía superior, pasa a pertenecer a la jerarquía superior B, el modelo puede advertir o no (según la configuración elegida) que el elemento de nivel inferior en cuestión ha cambiado de padre, y permitir o no ese cambio.

En general, se trata de una ventana que ofrece la mayor configuración de los modelos y una vista muy significativa del modelo general de un cubo.

Orígenes de datos (Data Sources)

En la ventana se muestran los orígenes de datos configurados para el modelo. No existe restricción en la cantidad o la diversidad de tipo de orígenes de datos. Cada origen de datos o Data Source incorpora una o varias, por lo general, consultas independientes, que serán el origen de las dimensiones y los hechos.

Todo dato, etiqueta, código de ordenación, código de categorización[1], o elemento utilizado en algún cálculo de medidas debe estar definido como parte de un Data Source especificado.

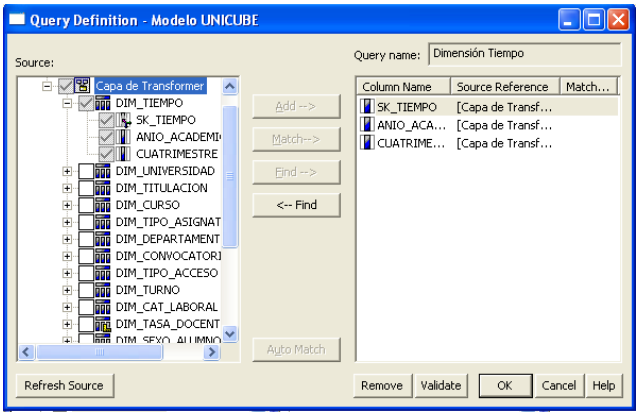


Ilustración 34. Selección Origen de datos desde el paquete publicado

[1] Category Code. Se trata de una clave única para cada elemento de un cubo dimensional, equivalentes a las claves unívocas a nivel de tabla en base datos.

Medidas (Measures)

En esta ventana se establecen los atributos de hechos del modelo. Declarados en el Data Source. Es importante destacar que en la definición de origen de datos dentro de Transformer no se establece que será hecho o que será dimensión, es el uso de ese origen de datos lo que define que es.

Las medidas pueden ser directamente un hecho que se agrupa utilizando los operadores dimensionales definidos o una medida calculada a partir de otras medidas declaradas.

No todas las medidas declaradas tienen por qué ser visibles en un cubo, en ocasiones, algunas funcionan como auxiliares para el cálculo de un atributo de hecho que se desea mostrar.

Cubos (PowerCubes)

Un modelo puede alimentar a un número indefinido de cubos. Cada cubo es configurado para mostrar y leer de unas determinadas dimensiones, y por tanto de un origen de datos, mostrar unos u otros atributos de hechos y ser publicados bajo un nombre y en una localización de Cognos Connection independiente.

Vistas personalizadas (Custom Views)

Las vistas personalizadas permiten a un cubo del modelo en concreto visualizar parcialmente dimensiones y seleccionar atributos de hechos.

La diferencia entre una vista personalizada y un cubo, es que con las vistas se puede ver una dimensión parcialmente, es decir, un cubo tiene o no tiene una dimensión, una vista personalizada tiene la dimensión y ve solo parte de sus elementos, aquellos para los que tiene permisos.

Por tanto las vistas se deben asociar a usuarios que las visualicen. Esto es de utilidad cuando en grandes compañías los usuarios con acceso a un cubo tienen perfiles diferentes y no pueden acceder a la información global. En estos casos se implementa seguridad en el cubo y se accede a vistas personalizas.

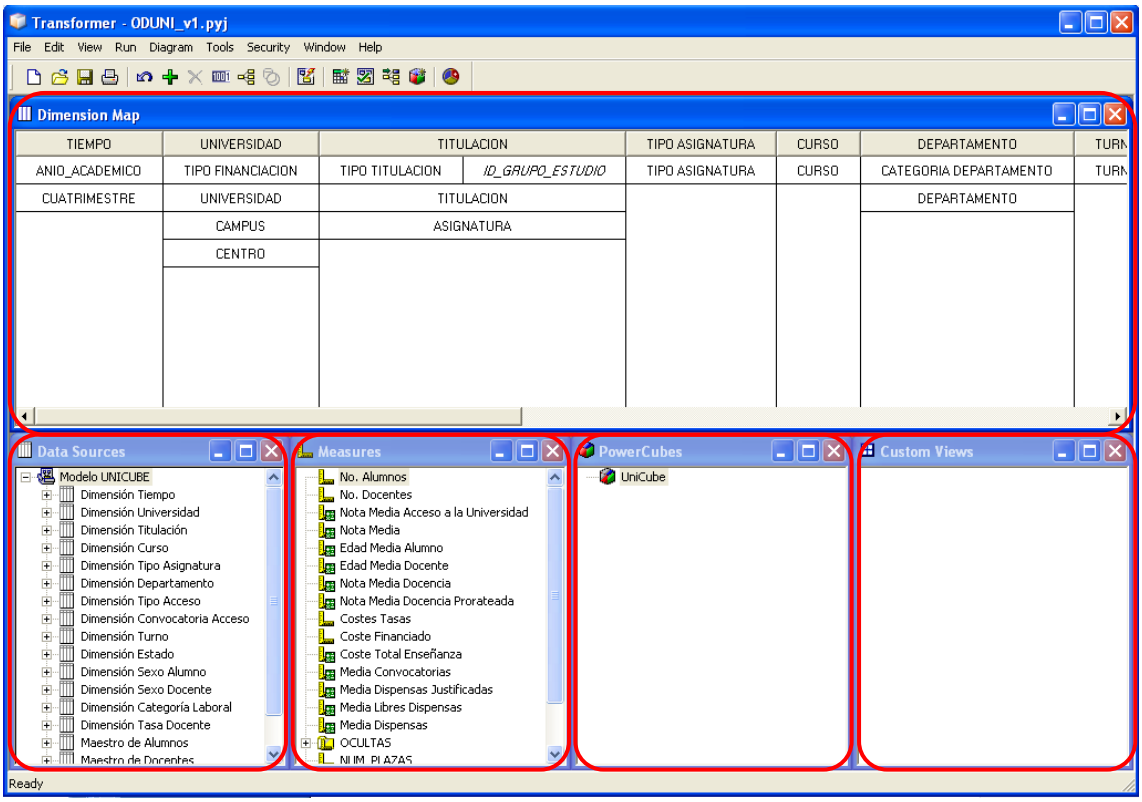


Ilustración 35. Ventana principal Transformer

Categorías (Categories)

Las categorías son los elementos de las dimensiones y sus jerarquías, es decir el año académico 2003/04 es una categoría del nivel año académico de la dimensión tiempo, el año 2005/06 es otra categoría.

Con la ventana de categoría se puede ver el árbol de las dimensiones y sus niveles con valores. Cabe destacar que las categorías de las dimensiones se alimentan de las tablas de dimensiones y que una categoría visible en esta ventana puede no aparecer posteriormente en el cubo, si ninguna de las medidas asociada a ellas muestra valores, es decir si no existen hechos para la categoría.

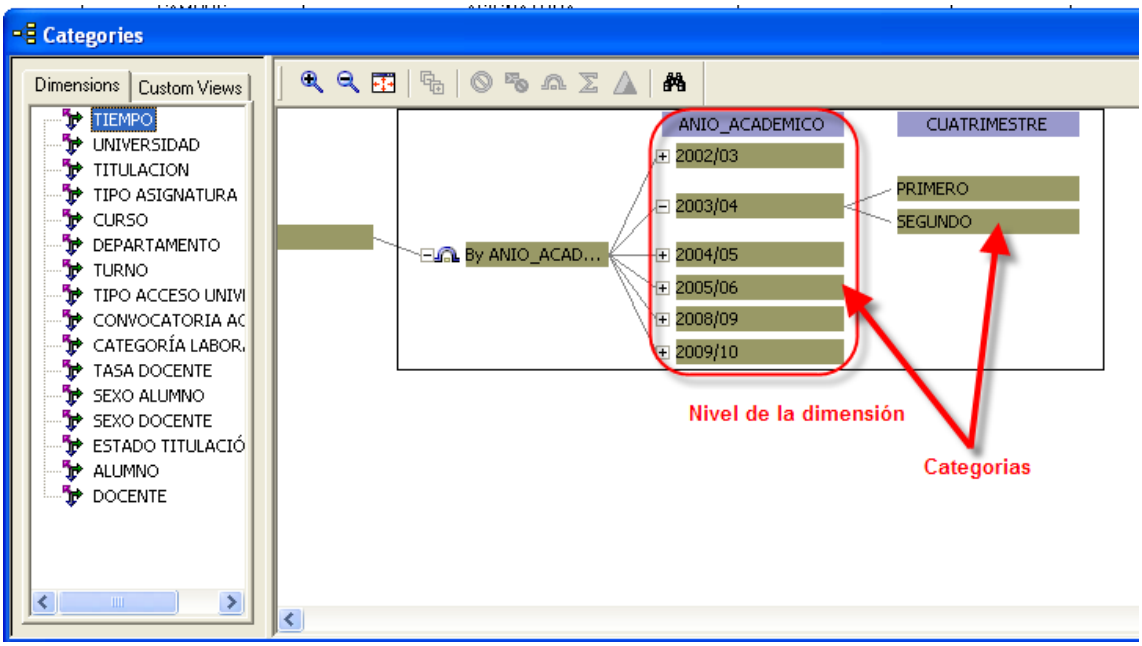


Ilustración 36. Ventana Categorías

Otros componentes de modelado que no se tratarán en el proyecto son:

3.3.2.3. Map Manager

Utilidad de Windows orientada a administradores y modeladores para importar mapas y actualizar etiquetas de mapas en la herramienta Report Studio.

Para las funciones de mapa como los nombres de ciudades y países, se pueden definir nombres alternativos para ofrecer versiones multilingües del texto.

3.3.2.4. Metric Designer

Herramienta de modelado complementaria a Metric Studio. Se emplea para crear extracciones, donde se asigna y transfiere información de los orígenes de metadatos existentes, y utilizarlas en las aplicaciones de tablas de puntuación (creadas con Metric Studio).

3.3.2.5. Series 7 IQD Bridge

Componente para la compatibilidad entre los orígenes de datos de IBM Cognos Series 7 Impromptu IQD y las consultas externalizadas de Framework Manager en Transformer.

3.3.3. Otros componentes

Además de las herramientas proporcionadas con IBM Cognos 8, se necesitan los siguientes componentes adicionales que se crean con otros recursos.

3.3.3.1. Almacén de contenido

El almacén de contenido es una base de datos relacional que contiene los datos que necesita Cognos 8 para funcionar, como especificaciones de informes, modelos publicados y los paquetes que los contienen; información de conexión de orígenes de datos; información sobre el espacio de nombres externo y el propio espacio de nombres de Cognos, e información sobre informes de separación y planificación.

La base de datos del Almacén de Contenido debe crearse utilizando una de las bases de datos que figuran en la siguiente tabla:

Base de datos	Codificación de caracteres	Protocolo
DB2	UTF-8	TCP/IP
Oracle	AL32UTF8 o AL32UTF16	TCP/IP
SQL Server	UTF-8 o UTF-16	TCP/IP
Sybase	UTF-8	TCP/IP
Base de datos de contenido de Cognos	Pre-configurada	Pre-configurada

Por restricciones de licencias y por menor complejidad en la configuración se ha optado por una base de datos Microsoft SQL Server. A la base de datos creada se la ha llamado CM.

Con Query Analizer, programa editor para SQL Server 2000 puede crearse la base de datos y visualizarla, así como los objetos que se definen de forma automática conforme se da uso a Cognos.

Sentencia SQL para la creación del Almacén de Contenido
Create database CM;

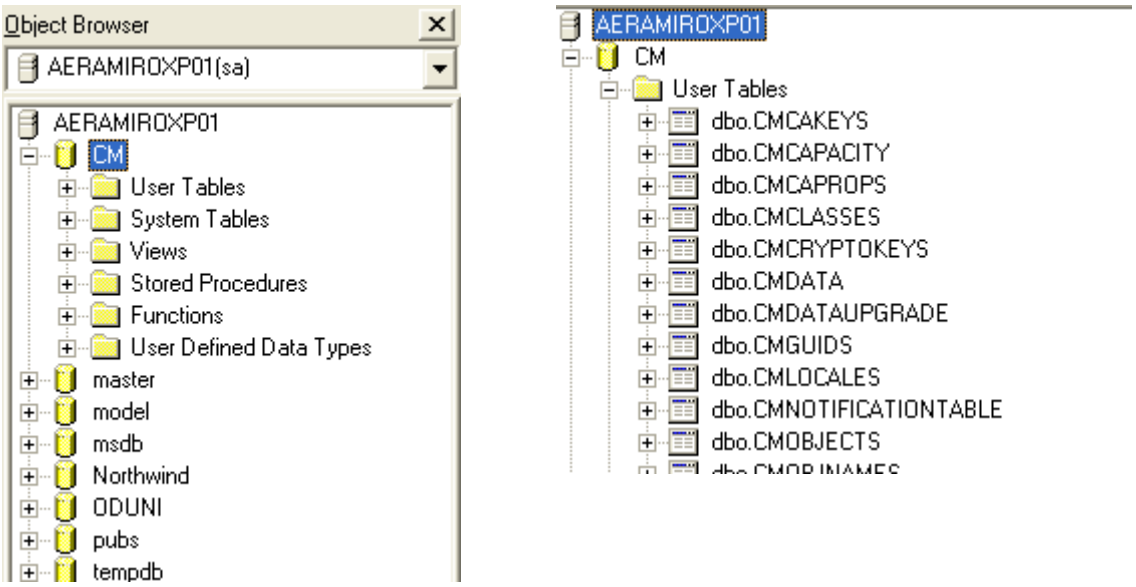


Ilustración 37. Vista base de datos CM

No es preciso configurar esta base de datos, ya que su gestión es interna de Cognos y tranparente al usuario. Una vez creado, el almacén de contenido en SQL Server debe añadirse el recurso del Content Manager a través de IBM Cognos Configuration, ya que por defecto, se configura un Almacén de Contenido en DB2.

Como el tipo de base de datos no es un valor modificable y no se puede añadir un recurso extra porque hay componentes, como este mismo, que sólo pueden utilizar un recurso al mismo tiempo, antes de crear otro recurso en el Content Manager, debe eliminarse el recurso predefinido.

3.3.3.2. Orígenes de datos

También conocidos como bases de datos de consultas, son bases de datos relacionales, cubos dimensionales, archivos u otros almacenes de datos físicos a los que se puede acceder a través de Cognos 8.

Los componentes de nivel de aplicación utilizan conexiones de orígenes de datos para acceder a los orígenes de datos.

En la práctica, se ha creado y configurado una base de datos bajo un modelo E/R, bautizada como ODUNI (Origen de Datos de las UNIversidades) en SQL Server 2000.

Sentencia SQL para la creación de la base de datos
Create database ODUNI;

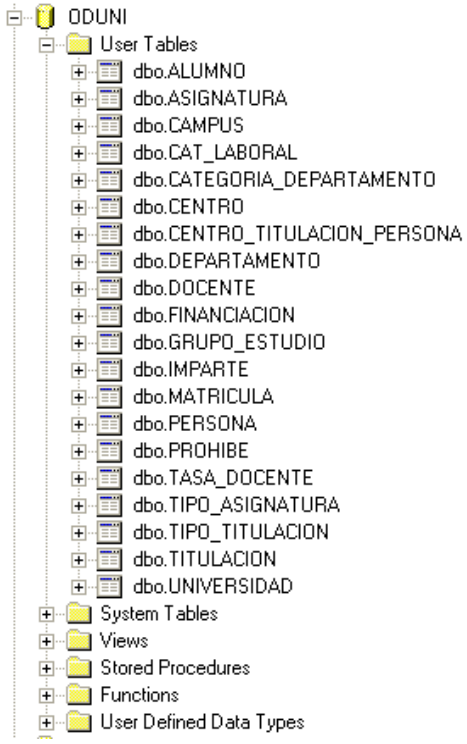


Ilustración 38. Vista Base de datos ODUNI

La elección del SGBDR SQL Server 2000 se ha debido a dos causas fundamentales:

- Ya existe una conexión entre SQL Server 2000 y Cognos, debido al Content Manager, quien se vale de una base de datos definida en SQL Server para almacenar información interna de Cognos.

- Una base de datos en este programa requiere una configuración más sencilla y con menos probabilidades de errores de conexión.

El origen de datos ODUNI es explotado por el componente Framework Manager, para realizar la conexión entre ambos se ha valido de un Origen de Datos ODBC (en castellano, Conectividad Abierta de Base de Datos), administrado a través de una herramienta administrativa incluida por Windows XP. Los pasos seguidos para la configuración de este origen de datos pueden leerse en el Anexo ¿?

Otro tipo de componente, fuera del alcance de este estudio.

3.3.3.3. Almacén de indicadores

Se trata de un componente estrechamente relacionado con los CM y los componentes Metric Studio y Metric Designer.

Un almacén de indicadores es una base de datos relacional que incluye contenido para paquetes de indicadores, además contiene la configuración de Metric Studio. Puede coexistir más de un almacén de indicadores.

3.4. Herramienta Cognos 8. Navegadores predefinidos

Para la elaboración de un *cubo multidimensional* se utilizan diferentes componentes de Cognos 8, sin embargo las herramientas donde se modela y se analiza son las que presentan los *operadores* sobre *cubos* o *tipos de navegación multidimensional*, estas son Transformer y Analisys Studio.

A continuación, se tratarán las consultas predefinidas del modelo multidimensional dinámico que incorporan estos componentes.

3.4.1. Exploración Descendente: Drill-down

Navegación desde los datos más resumidos o agregados hasta el detalle de estos. En las dimensiones se trata de descender niveles en la jerarquía definida.

3.4.2. Exploración Ascendente: Roll-up

Navegación desde los valores más detallados hasta los valores agregados que los contienen. Consiste en ascender los niveles de la jerarquía de las dimensiones.

3.4.3. Proyectar: Slicing y Dicing

Consiste en fijar los valores de las dimensiones en una tabla, obteniendo una “rebanada” de la BD.

3.4.4. Pivotar: Pivoting

Pivotando se transpone la representación del esquema de hecho, rotando la visión multidimensional de los datos sin que estos se alteren.

3.4.5. Anidamiento: Nesting

Es la visualización de un *sub-cubo*, el resultado de una consulta en las que interponen varias dimensiones.

3.4.6. Extensión: Reach Through

Permita la consulta y recuperación automática de los datos del DWH.

3.5. Herramienta Cognos 8. Operadores predefinidos

De la misma manera que los *tipos de navegación*, los *operadores multidimensionales* aplican a los componentes Transformer y Analisys Studio.

En los *operadores* el Framework manager, tiene un papel fundamental permitiendo aplicar funciones a los datos que desde el modelado en Transformer no es posible. Además se pueden especificar los tipos de agregación de los datos del modelo que luego serán datos de origen para Transformer.

Las funciones de agregación principales se encuentran disponibles en las herramientas:

3.5.1. Distributivas

Operador	Funcionalidad
Suma	Suma los valores de todos los elementos del conjunto activo.
Mínimo	Devuelve el menor valor de entre todos los elementos del conjunto activo.
Máximo	Devuelve el mayor valor de entre todos los elementos del conjunto activo.
Contar	Cuenta los valores no nulos de todos los elementos del conjunto activo, los valores repetidos contabilizan tantas veces como estos se repitan.

3.5.2. Algebraicas

Operador	Funcionalidad
Media	Calcula la media aritmética de los valores del conjunto activo.
Varianza	Devuelve la varianza de los elementos de datos seleccionados.
Desviación standard	Devuelve la desviación estándar de los elementos de datos seleccionados.
Absoluto	Devuelve el valor absoluto del elemento.
Porcentaje	Retorna el porcentaje del valor del elemento en función de otro especificado.
Standard deviation- pop	Calcula la desviación estándar de la población y devuelve la raíz cuadrada de la varianza de la población.

3.5.1. Holísticas

Operador	Funcionalidad
Mediana	Devuelve la mediana de los elementos de datos seleccionados.
Rank	Devuelve el valor de rango de los elementos de datos seleccionados. De forma predeterminada, aparecen en orden descendente con los valores nulos al final.
Percentil	Devuelve un valor, en escala de cien, que indica el porcentaje de una distribución igual o menor que los elementos de datos seleccionados.

Cant.	Percentil (Cant.)
800	1
700	0,9
600	0,8
500	0,6

400	0,5
400	0,5
200	0,3
200	0,3

Quantile

Devuelve el rango de un valor para un rango especificado. Devuelve números enteros para representar cualquier rango de posiciones.

Cant.	Rango (Cant.)	Cuantil (Cant.; 4)
800	1	1
700	2	1
600	3	2
500	4	2

400	5	3
400	5	3
200	7	4
200	7	4

Quartile

Devuelve el rango de un valor, representado como números enteros del 1 al 4 en relación a un grupo de valores.

Cant.	Cuartil (Cant.)
450	1
400	1
350	2
300	2

250	3
200	3
150	4
100	4

Running average

Devuelve la media parcial por fila de un conjunto de valores.

Nombre	Cant.	Media	Media parcial
Smith	7	5	7
Smith	3	5	5
Smith	6	5	5,33
Smith	4	5	5
Wong	3	4	3
Wong	5	4	4

Running count

Devuelve el recuento parcial por fila de un conjunto de valores.

Nombre	Cant.	Recuento	Recuento parcial
Smith	7	4	1
Smith	3	4	2
Smith	6	4	3
Smith	4	4	4
Wong	3	3	1
Wong	5	3	2

3.6. Estudio de alternativas

Con objeto de realizar el caso práctico, el análisis de otras vías se dividirá en tres partes fundamentales :

- Como primer punto aparece el desarrollo de una BD relacional.

  Para este apartado se ha optado por una aplicación SQL Server 2000. Se trata de una base de datos de fácil configuración para trabajar con Cognos 8, programa al que se liga durante todo el desarrollo.

  Existen programas menos especializados en el tratamiento masivo de datos, como puede ser Microsoft Access, MySQL que tienen como principal limitación el volumen de datos capaz de procesar. U Oracle, cuya tecnología permite la recuperación de un volumen mayor de datos.

- En segundo lugar, se encuentra la necesidad de realizar una *ETL*, que nos lleve los datos desde el modelo relacional hasta los *Datamart*, organizados en un *esquema en estrella*, que forman la *DWH*.

  PowerCenter se presenta como una muy buena alternativa, si bien el coste de las licencias, necesarias para el uso de esta herramienta, hacen imposible seleccionarlo para realizar la *ETL*.

  Por ello, este paso en el proceso se realiza a través de la creación secuencial de vistas en el mismo motor de base de datos SQL Server 2000, donde se aloja la *BBDDR*.

  Esta alternativa, menos profesional, es posible porque se trata de un caso práctico con una sola base de datos relacional como origen con una cantidad de datos alta pero manejable para el motor, que sin ser el más eficiente para estos desarrollos puede usarse en esta ocasión concreta.

- Por último, para la explotación del sistema de *BI* construido se utiliza el paquete de IBM Cognos 8. El programa Transformer, contenido en dicho paquete, permite la creación de potentes *cubos* de forma intuitiva, su interfaz resulta amigable para el usuario desarrollador y final analista.

  Además ofrece la posibilidad de aumentar el alcance del proyecto en un futuro, debido a la gran variedad de componentes de la herramienta, que otros software no incorporan, como la capacidad de aumentar los orígenes de datos con otras bases de datos, la posibilidad de realizar informes a través del Report Studio, indicadores y Cuadros de Mando con Metric Designer y Metric Studio.

Otros programas muy competentes resultan alternativas técnicamente viables. Sin embargo, el alto coste de las licencias imposibilita su uso en este proyecto. Las licencias de Cognos 8 han sido cedidas, lo que ha permitido el desarrollo con esta aplicación.

3.7. Plan de proyecto software

En el plan de proyecto se detalla la planificación temporal, necesaria para su desarrollo, y el presupuesto disponible.

En el desarrollo intervendrán los siguientes roles:

- Diseñador: Responsable del modelado de los almacenes de datos y transformaciones entre esquemas.

- Programador: Implementador del sistema definido por el diseñador. Debe ajustarse al diseño concebido y realizar las funciones detalladas en el diseño.

- Encargado de pruebas: Encargado de verificar la correcta conversión de los datos en los modelos definidos e implementados. Los datos de prueba que establezca deben asegurar la validez del sistema.

- Gestor: Enlaza al resto de roles y coordina las acciones que deben llevarse a cabo.

Los tres primeros roles serán asumidos por Ana Ramiro y la gestión del trabajo será supervisada por la tutora del proyecto, Elena Castro y la directora del mismo Jessica Rivero.

Se realizarán una serie de reuniones de seguimiento del proyecto a lo largo del desarrollo de este para ir comprobando que la ejecución del proyecto se ajusta a lo esperado. Estas reuniones tendrán lugar entre el gestor y resto de roles.

El ciclo de vida del proyecto consistirá fundamentalmente en una serie de fases en cascada:

- Estudio general del caso: Se analizará el contexto en el que se desarrolla el proyecto y se especificará el alcance de éste.

- Planificación y estudio de la viabilidad: Se hará un estudio de las herramientas a utilizar y se profundizará en el manejo de las seleccionadas para desarrollar el proyecto.

  En esta fase se establecerá la planificación temporal y presupuestaria de todo el proyecto, haciendo una estimación del esfuerzo y costes que serán necesarios.

- Análisis y diseño: Fase en la que se desarrolla el proyecto a nivel conceptual. Se establecen los modelos de datos y las estructuras a implementar.

- Implementación y pruebas: En este punto se lleva a cabo la implementación del sistema y se realizan las pruebas que lo testean. Se trata de una fase en la que se repite varias veces la implementación y las posteriores pruebas, hasta perfeccionar el sistema.

El proyecto se organizará en función de las cuatro fases descritas, para representar gráficamente el trabajo se ha empleado un diagrama WBS (Work Breakdown Structure) que puede verse en la Ilustración 39. La numeración entre paréntesis, anterior al nombre de cada actividad elemental en la identificación establecida en el diagrama GANTT realizado posteriormente, véase Ilustración 40.

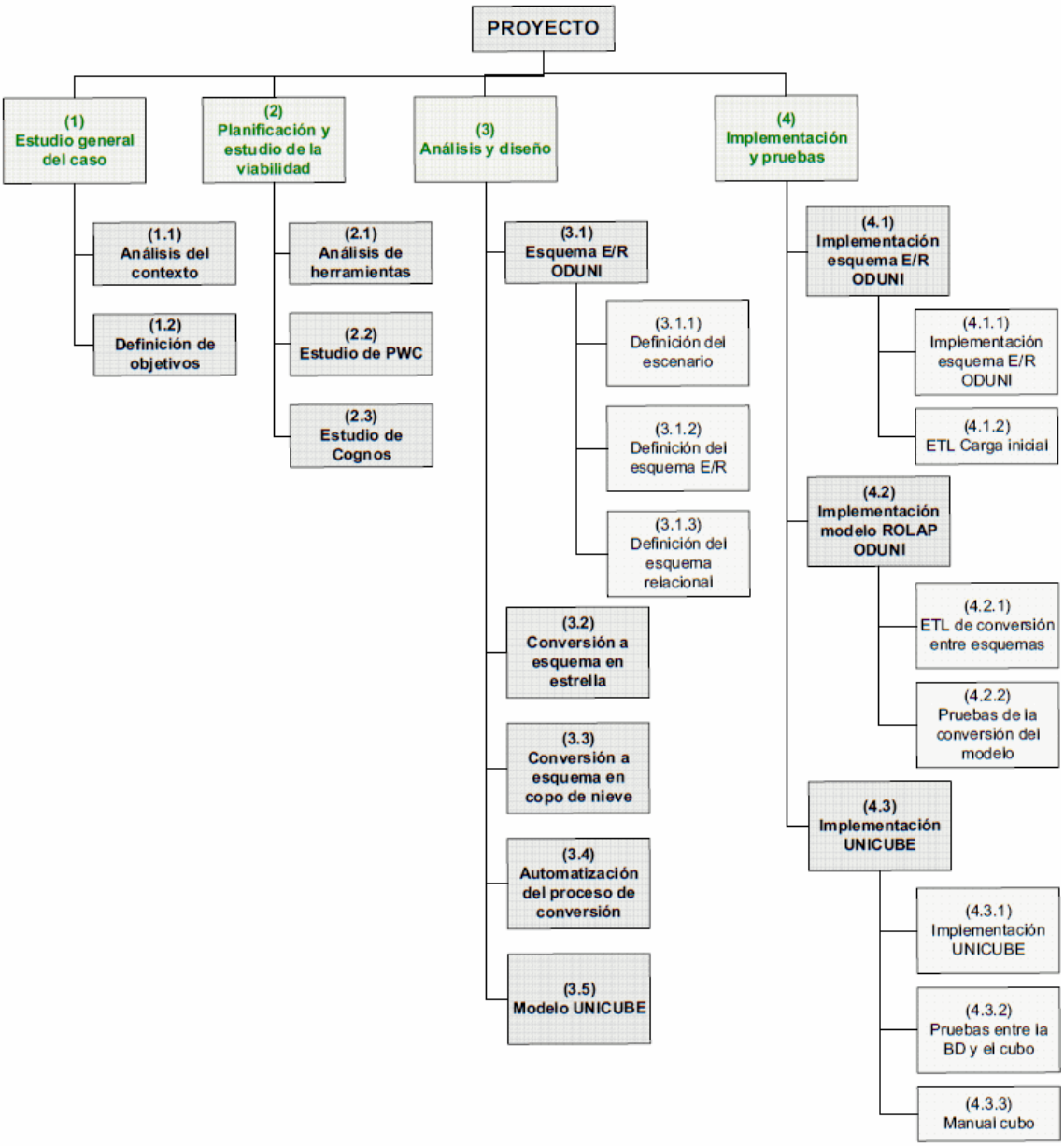


Ilustración 39. Estructura de tareas WBS

El proyecto tendrá una duración aproximada de 7 meses, comenzando el día 4 de octubre del 2011 y finalizando el 1 de mayo de 2012.

El calendario laboral consta de:

- Cinco días laborables a la semana.
- Ocho horas de trabajo diarias.

- Días festivos en el periodo de realización del proyecto: 1 de noviembre, 6 y 8 de diciembre, 6 de enero, 19 de marzo, 5 y 6 de abril.

ID	TAREA	DURACIÓN	FECHA INICIO	FECHA FIN
1	Proyecto	146 días	04/10/2011	01/05/2012
2	(1) Estudio general del caso	11 días	04/10/2011	18/10/2011
3	(1.1) Análisis del contexto	6 días	04/10/2011	11/10/2011
4	(1.2) Definición de objetivos	5 días	12/10/2011	18/10/2011
5	(2) Planificación y estudio de la viabilidad	35 días	19/10/2011	12/12/2011
6	(2.1) Análisis de herramientas	5 días	19/10/2011	25/10/2011
7	(2.2) Estudio de PWC	17 días	26/10/2011	18/11/2011
8	(2.3) Estudio de Cognos	26 días	02/11/2011	12/12/2011
9	(3) Análisis y diseño	66 días	13/12/2011	14/03/2012
10	(3.1) Esquema E/R ODUNI	22 días	13/12/2011	12/01/2012
11	(3.1.1) Definición del escenario	11 días	13/12/2011	27/12/2011
12	(3.1.2) Definición del esquema E/R	4 días	28/12/2011	02/01/2012
13	(3.1.3) Definición del esquema relacional	7 días	03/01/2012	12/01/2012
14	(3.2) Conversión a esquema en estrella	8 días	03/02/2012	14/02/2012
15	(3.3) Conversión a esquema en copo de nieve	5 días	22/02/2012	28/02/2012
16	(3.4) Automatización del proceso de conversión	6 días	27/02/2012	05/03/2012
17	(3.5) Modelo UNICUBE	5 días	08/03/2012	14/03/2012
18	(4) Implementación y pruebas	34 días	15/03/2012	01/05/2012
19	(4.1) Implementación esquema E/R ODUNI	11 días	15/03/2012	29/03/2012
20	(4.1.1) Implementación esquema E/R ODUNI	3 días	15/03/2012	19/03/2012
21	(4.1.2) ETL Carga inicial	8 días	20/03/2012	29/03/2012
22	(4.2) Implementación modelo ROLAP ODUNI	10 días	02/04/2012	13/04/2012
23	(4.2.1) ETL de conversión entre esquemas	5 días	02/04/2012	06/04/2012
24	(4.2.2) Pruebas de la conversión del modelo	5 días	09/04/2012	13/04/2012
25	(4.3) Implementación UNICUBE	12 días	16/04/2012	01/05/2012
26	(4.3.1) Implementación UNICUBE	3 días	16/04/2012	18/04/2012
27	(4.3.2) Pruebas entre la BD y el cubo	6 días	20/04/2012	27/04/2012
28	(4.3.3) Manual cubo	2 días	30/04/2012	01/05/2012

Tabla 1. Tareas Gantt

	Nombre de tarea	Duración	Comienzo	Fin
1	**Proyecto**	**146 días**	**mar 04/10/11**	**mar 01/05/12**
2	**(1) Estudio general del caso**	**11 días**	**mar 04/10/11**	**mar 18/10/11**
3	(1.1) Analisis del contexto	6 días	mar 04/10/11	mar 11/10/11
4	(1.2) Definición de objetivos	5 días	mié 12/10/11	mar 18/10/11
5	**(2) Planificación y estudio de la viabilidad**	**35 días**	**mié 19/10/11**	**lun 12/12/11**
6	(2.1) Analisis de herramientas	5 días	mié 19/10/11	mar 25/10/11
7	(2.2) Estudio de PWC	17 días	mié 26/10/11	vie 18/11/11
8	(2.3) Estudio de Cognos	26 días	mié 02/11/11	lun 12/12/11
9	**(3) Analisis y diseño**	**66 días**	**mar 13/12/11**	**mié 14/03/12**
10	**(3.1) Esquema E/R ODUNI**	**22 días**	**mar 13/12/11**	**jue 12/01/12**
11	(3.1.1) Definición del escenario	11 días	mar 13/12/11	mar 27/12/11
12	(3.1.2) Definición del esquema E/R	4 días	mié 28/12/11	lun 02/01/12
13	(3.1.3) Definición del esquema relacional	7 días	mar 03/01/12	jue 12/01/12
14	(3.2) Conversión a esquema en estrella	8 días	vie 03/02/12	mar 14/02/12
15	(3.3) Conversión a esquema en copo de nieve	5 días	mié 22/02/12	mar 28/02/12
16	(3.4) Automatización del proceso de conversión	6 días	lun 27/02/12	lun 05/03/12
17	(3.5) Modelo UNICUBE	5 días	jue 08/03/12	mié 14/03/12
18	**(4) Implementación y pruebas**	**34 días**	**jue 15/03/12**	**mar 01/05/12**
19	**(4.1) Implementación esquema E/R ODUNI**	**11 días**	**jue 15/03/12**	**jue 29/03/12**
20	(4.1.1) Implementación esquema E/R ODUNI	3 días	jue 15/03/12	lun 19/03/12
21	(4.1.2) ETL Carga inicial	8 días	mar 20/03/12	jue 29/03/12
22	**(4.2) Implementación modelo ROLAP ODUNI**	**10 días**	**lun 02/04/12**	**vie 13/04/12**
23	(4.2.1) ETL de conversión entre esquemas	5 días	lun 02/04/12	vie 06/04/12
24	(4.2.2) Pruebas de la conversión del modelo	5 días	lun 09/04/12	vie 13/04/12
25	**(4.3) Implementación UNICUBE**	**12 días**	**lun 16/04/12**	**mar 01/05/12**
26	(4.3.1) Implementación UNICUBE	3 días	lun 16/04/12	mié 18/04/12
27	(4.3.2) Pruebas entre la BD y el cubo	6 días	vie 20/04/12	vie 27/04/12
28	(4.3.3) Manual cubo	2 días	lun 30/04/12	mar 01/05/12

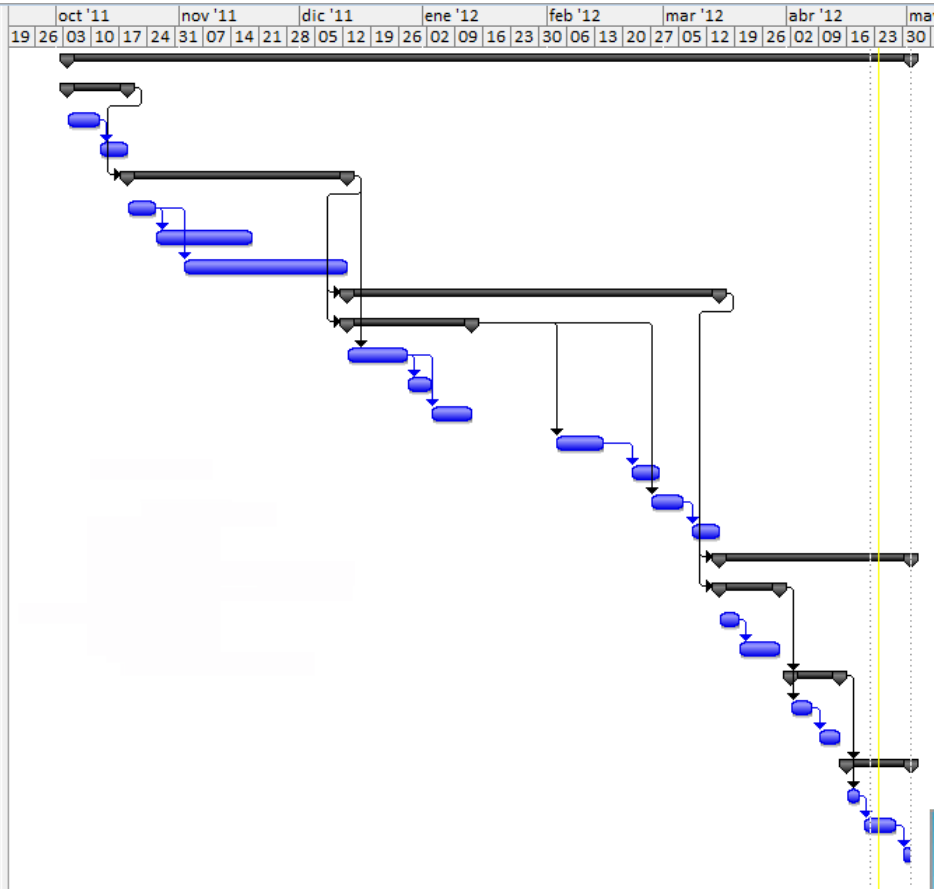


Ilustración 40. Diagrama de GANTT

UNIVERSIDAD CARLOS III DE MADRID
Escuela Politécnica Superior

PRESUPUESTO DE PROYECTO

1.- Autor:

Ana Esther Ramiro Bonilla

2.- Departamento:

Ingeniería Técnica Informática de Gestión

3.- Descripción del Proyecto:

Titulo	**MODELO MULTIDIMENSIONAL: ANALISIS DE MECANISMOS DE EXTRACCIÓN DEL CONOCIMIENTO A PARTIR DE ESQUEMAS EER**
Duración	**8 meses**
Tasa de costes Indirectos:	**20%**

4.- Presupuesto total del Proyecto (valores en Euros):

Euros

5.- Desglose presupuestario (costes directos)

PERSONAL

Apellidos y nombre	N.I.F. (no rellenar - solo a título informativo)	Categoría	Dedicación (hombres mes) [a)]	Coste hombre mes	Coste (Euro)
Ana Esther Ramiro Bonilla		Diseñador	6,82	4.593,75 €	31.329,38
Ana Esther Ramiro Bonilla		Programador	1,40	2.625,00 €	3.675,00
Ana Esther Ramiro Bonilla		Encargado de pruebas	0,67	3.937,50 €	2.638,13
Elena Castro Galán		Gestor	2,22	6.562,50 €	21.887,73
Jessica Rivero Espinosa		Gestor	1,66	6.562,50 €	7.292,58
		Hombres mes	**13,335**	**Total**	**66.812,81**

[a)] 1 Hombre mes = 131,25 horas. Máximo anual de dedicación de 12 hombres mes (1.575 horas)

Máximo anual para PDI de la Universidad Carlos III de Madrid de 8,8 hombres mes (1.155 horas)

EQUIPOS

Descripción	Coste (anual/Euro)	% Uso dedicado proyecto	Dedicación (meses)	Periodo de depreciación	Coste imputable [d)]
Renting Ordenador Portátil	600,00	100	8	12	400,00
Renting Licencia Windows XP	0,00	100	8	12	0,00
Renting Licencia SQL Server 2000	2.500	100	8	12	1.666,67
Renting Licencia Cognos 8	3.300,00	100	8	12	2.200,00
				Total	**4.266,67**

[d)] Fórmula de cálculo de la Amortización:

$$\frac{A}{B} x C x D$$

A = nº de meses desde la fecha de facturación en que el equipo es utilizado

B = periodo de depreciación (60 meses)

C = coste del equipo (sin IVA)

D = % del uso que se dedica al proyecto (habitualmente 100%)

SUBCONTRATACIÓN DE TAREAS

Descripción	Empresa	Coste imputable
n.a.		
	Total	0,00

OTROS COSTES DIRECTOS DEL PROYECTO[e)]

Descripción	Empresa	Costes imputable
n.a.		
	Total	0,00

[e)] Este capítulo de gastos incluye todos los gastos no contemplados en los conceptos anteriores, por ejemplo: fungible, viajes y dietas, otros,…

6.- Resumen de costes (euros)

Presupuesto Costes Totales	**Presupuesto Costes Totales**
Personal	66.812,81
Equipos	4.266,67
Subcontratación de tareas	0,00
Costes de funcionamiento	0,00
Costes Indirectos	0,00
Total	**71.079,48 €**

Capítulo 4

En este capítulo se analizará la metodología del proyecto. Tendrán lugar el diseño del esquema *E/R*, el diseño de la conversión a los *esquemas en estrella* y *copo de nieve* y se tratará de analizar la automatización de este tipo de conversiones.

4. ANÁLISIS Y DISEÑO DEL SISTEMA

Una vez descrito un sistema de *BI* y las herramientas, que serán necesarias para cumplir el objetivo del proyecto, de analizar la conversión entre modelos de datos y analizar un modelo multidimensional diseñado en un *esquema en estrella*, se procederá en el presente capitulo a definir la metodología, analizar el caso concreto y diseñar el sistema.

El esquema de datos inicial *E/R* versará sobre la información de Universidades en lo que se refiere las titulaciones, a los alumnos que las cursan y los profesores que las imparten. En modelo estará implementado sobre una BD relacional. A esta BD se le ha llamado ODUNI (Origen de Datos de las UNIversidades)

Mediante una *ETL* se procederá a la carga de información, ficticia, del modelo inicial.

Posteriormente, se estudiará la conversión del esquema inicial *E/R* en dos modelos utilizados para análisis multidimensionales: *esquema en estrella* y *esquema en copo de nieve*.

Se procederá a la conversión del esquema *E/R* a un *esquema en estrella* a través de una segunda *ETL* (distinta de la primera) que contenga la misma información, correcta y completa, que el primer modelo. Esta transformación se llevará a cabo con el mismo *GBDR* en donde se define la base de datos *E/R*.

El *esquema en estrella*, al ser un modelo de datos multidimensional, su gestión más optima sería a través de un *Sistema Gestor de Bases de Datos Multidimensional* (*SGBDMD*), sin embargo estos modelos no está asociado a una representación física de los datos por lo que pueden estar soportados en un *SGBDR*, perdiendo, eso sí, eficiencia. A estos modelos de datos se les llama *ROLAP* (*Relational On-Line Analytical Processing*).

Una vez la información se aloja en un modelo de datos multidimensional puede ser explotados por herramientas *OLAP*, así se procederá a la definición de un *cubo* en Cognos 8. Se ha bautizado al *cubo* con el nombre UNICUBE.

En este capítulo, por tanto, se definirá el modelo ODUNI, los *esquemas en estrella* y en *copo de nieve* resultantes de transformar ODUNI y el *cubo* UNICUBE.

4.1. Esquema E/R ODUNI

La información a recopilar para desarrollar el caso práctico hará referencia a la Universidad como organización pública o privada, las titulaciones, los alumnos y los profesores. El modelo de datos deberá poder recoger de la forma más óptima y lógica la información necesaria. La integridad y consistencia de los datos queda asegurada, sin

detallarse en este escrito los pasos seguidos para conseguirla, ya que el objetivo es la conversión de modelos manteniendo los datos y no los datos en sí.

Lógicamente cada Universidad cuenta con su propio catálogo de titulaciones, tiene un número de campus y una normativa específica. Cada facultad entonces tendría sus aplicaciones particulares y un modelo de datos.

¿Por qué entonces una única BD y un único esquema *E/R* cómo origen de datos? En este caso práctico, se parte de una información homogeneizada para las diferentes facultades. Esto no implica que tengan la misma normativa, sino que cumplen una estructura y una serie de requisitos generales. El detalle de la normativa no es objeto de estudio, así como los datos no son información, algunos obedecen a información de dominio público de las Universidades, los datos de carácter privado son ficticios (como las notas, o los alumnos y profesores)

El objetivo de la BD no son los datos que contiene si no el modelo que la define y que será convertido.

4.1.1. Descripción del escenario a modelar

Para este caso, existen una serie de Universidades, que pueden obtener una financiación pública o privada. Cada una se compone de uno o varios campus que a su vez se dividen en centros. De las universidades, campus y centros se conoce el nombre.

Cada centro, de cada campus, oferta una serie plazas a las titulaciones, cada año académico y convocatoria (ordinaria y extraordinaria), en la que se pueden matricular los alumnos.

Para cada titulación ofertada en un centro, un determinado año académico, existen tres notas de acceso, una para cada camino por el que el alumno puede optar a matricularse: prueba de acceso general por bachillerato, prueba por formación profesional o accediendo siendo mayor de 25 años.

Un alumno, sólo se puede matricular en una titulación ofertada por un centro una única vez en un año académico y dejará de estar matriculado cuando su expediente académico se cierre, bien porque ha finalizado los estudios, bien porque no existe posibilidad de concluirlos (extinción de la carrera, excede el número de convocatorias…).

Las titulaciones, de las que se conoce: el nombre, los créditos ECTS de que consta, la tasa del crédito ECTS que debe abonar el alumno, las convocatorias a examen máximas a cualquiera de las asignaturas que la componen, la fecha en la que se implanta y en la que deja de impartirse; pertenecen a un tipo de titulación y un grupo de estudios, se encuentran en un estado en relación al plan vigente de estudios (ya sea “vigente”, si el plan lo contempla, “en extinción” si no es así pero aún se imparte docencia o “extinguida” si ya no se imparte docencia).

Las titulaciones están compuestas por una serie de tipos de asignaturas, que son únicas de cada titulación. De cada asignatura se conoce el nombre, el número de créditos ECTS y el curso al que pertenecen.

Además una asignatura es gestionada por un único departamento. Los departamentos se engloban en una unidad ficticia, categoría de departamento, de ambas estructuras se conoce el nombre.

Las categorías de departamento surgen de la necesidad de comparar universidades que no tienen establecidos los mismos departamentos, ni la misma relación entre asignatura y departamento. De este modo, se pueden analizar departamentos entre universidades diferentes, puestos que todas tendrán en común la categoría de departamento.

Existen dos tipos de personas en este modelo, los alumnos y los docentes (por simplicidad, una persona solo podrá ser uno de los dos). De cada persona interesa el DNI que los identifique, el nombre y apellidos, la edad y su sexo. En caso de tratarse de un alumno se debe conocer su tipo de acceso a la universidad y nota con la que accedió. En el supuesto de que la persona fuera un docente interesa su categoría laboral en la Universidad y la tasa asociada a esta categoría.

Los alumnos pueden cursar las asignaturas correspondientes a la titulación en la que están matriculados. Las cursan en un determinado año académico, en un único cuatrimestre y turno dentro de dicho año. Se debe conocer la nota obtenida en cada convocatoria y las dispensas, justificadas e injustificadas de cada asignatura, así como si la asignatura se ha cursado o no como libre elección (sólo en cierto tipo de titulaciones que definen este tipo de asignaturas).

Al igual que los alumnos cursan asignaturas, los docentes las imparten en un año académico y para todos los cuatrimestres dentro de ese año. Un profesor sólo puede impartir asignaturas de un mismo departamento (que será el departamento al que el profesor pertenece). Los profesores obtienen una nota que evalúa su docencia para cada asignatura y año académico.

4.1.2. Esquema E/R

El esquema E/R establecido para ODUNI, que representa el escenario descrito, es el que figura en la Ilustración 39. Esquema E/R ODUNI siguiente:

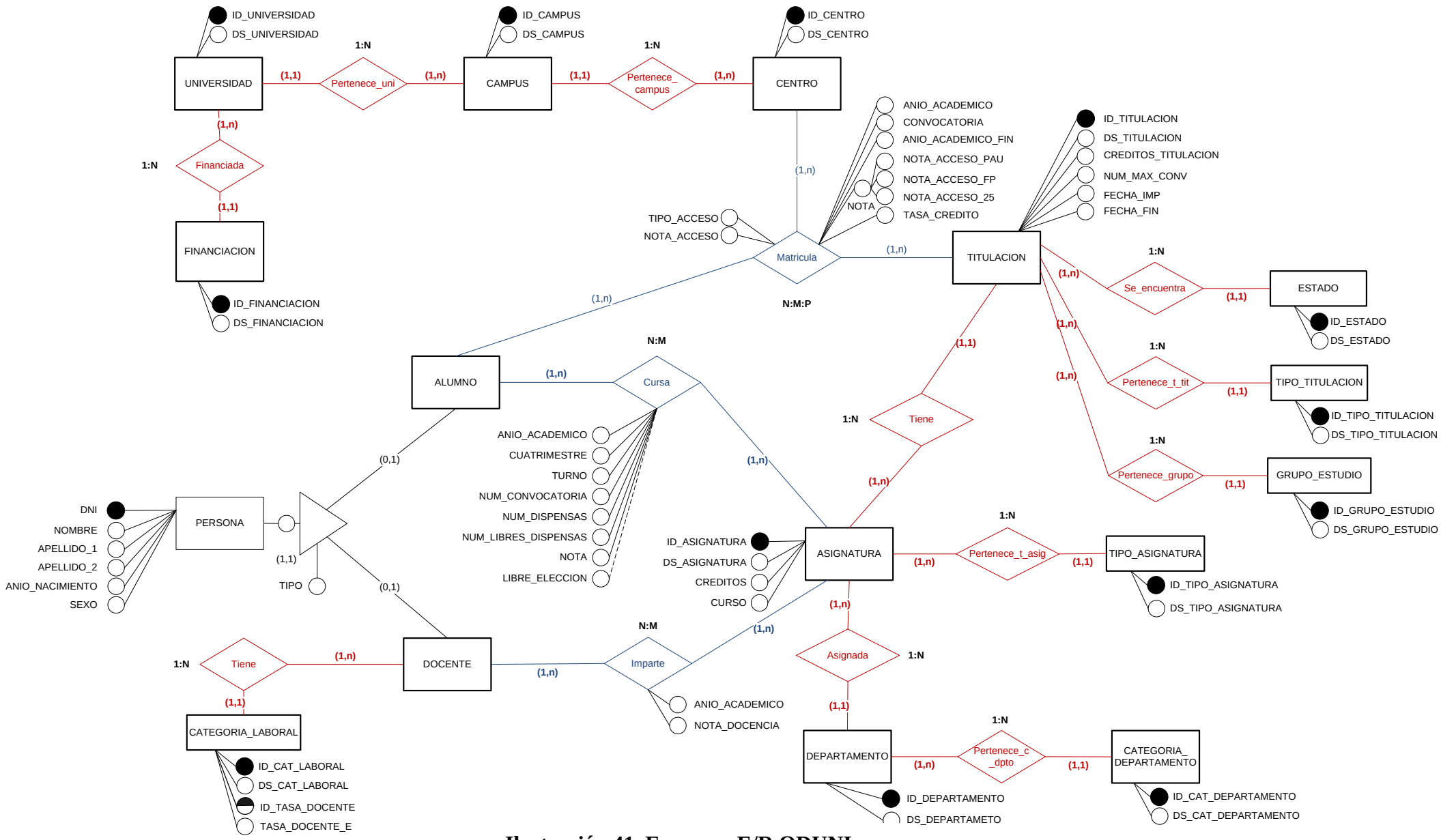


Ilustración 41. Esquema E/R ODUNI

A continuación, por orden alfabético, se describen las entidades con sus atributos:

ENTIDAD	ATRIBUTO	IDENTIFI-CADOR	OPCIONAL	DESCRIPCIÓN
ALUMNO	Forma parte de la jerarquía persona, por lo que hereda los atributos de su padre, PERSONA. La entidad carece de atributos propios, si bien establece relaciones que la obligan a establecerse como entidad.			
ASIGNATURA	Id_asignatura	SI	NO	Identificador de la asignatura
	Ds_asignatura	NO	NO	Nombre de la asignatura
	Creditos	NO	NO	Créditos ECTS de la asignatura
	Curso	NO	NO	Curso en que se imparte la asignatura
CAMPUS	Id_campus	SI	NO	Identificador del campus
	Ds_campus	NO	NO	Nombre del campus
CATEGORIA_ LABORAL	Id_cat_laboral	SI	NO	Identificador de la categoría laboral a la que puede pertenecer un profesor
	Ds_cat_laboral	NO	NO	Descripción de la categoría laboral
	Id_tasa_docente	A*	NO	Identificador de la tasa correspondiente a la categoría laboral
	Tasa_docente_E	NO	NO	Coste en Euros de la tasa
CATEGORIA_ DEPARTAMENTO	Id_cat_ departamento	SI	NO	Identificador de la categoría del departamento a la que puede pertenecer un departamento
	Ds_cat_ departamento	NO	NO	Descripción de la categoría del departamento

CENTRO	Id_centro	SI	NO	Identificador del centro de una universidad
	Ds_centro	NO	NO	Nombre del centro
DEPARTAMENTO	Id_departamento	SI	NO	Identificador del departamento al que puede pertenecer una asignatura
	Ds_departamento	NO	NO	Nombre del departamento
DOCENTE	Forma parte de la jerarquía persona, por lo que hereda los atributos de su padre, PERSONA. La entidad carece de atributos propios, si bien establece relaciones que la obligan a establecerse como entidad.			
ESTADO	Id_estado	SI	NO	Identificador del estado de un titulación
	Ds_estado	NO	NO	Descripción del estado. Puede ser "Vigente" cuando la titulación se encuentra en el plan de estudios vigente, "En extinción" si la titulación no pertenece al plan de estudios pero aún se imparte o "Extinguida" cuando deja de impartirse en la universidad
FINANCIACION	Id_financiacion	SI	NO	Identificador de la financiación de una universidad
	Ds_financiacion	NO	NO	Descripción de la financiación, puede ser "Pública" o "Privada"
GRUPO_ ESTUDIO	Id_grupo_ estudio		NO	Identificador del grupo de estudios al que pertenece una titulación
	Ds_grupo_ estudio	NO	NO	Nombre del grupo de estudios que obedece a la rama de la ciencia al que pertenece, humanidades y filosofía, ingeniería, historia…

PERSONA	DNI	SI	NO	DNI que identifica a cada persona
	Nombre	NO	NO	Nombre de la persona
	Apellido_1	NO	NO	Apellido principal de la persona
	Apellido_2	NO	NO	Segundo apellido de la persona
	Anio_nacimiento	NO	NO	Año de nacimiento de la persona
	Sexo	NO	NO	Género, hombre o mujer, de la persona
	Tipo	NO	NO	Especifica si la persona es un alumno o un docente
TIPO_ ASIGNATURA	Id_tipo_ asignatura	SI	NO	Identificador del tipo de asignatura que una dada puede ser
	Ds_tipo_ asignatura	NO	NO	Nombre del tipo de asignatura
TIPO_ TITULACION	Id_tipo_ titulacion	SI	NO	Identificador del tipo de titulación que una dada puede ser
	Ds_tipo_ titulacion	NO	NO	Descripción del tipo de titulación, grado, ingeniería, ingeniería técnica, licenciatura…
TITULACION	Id_titulacion	SI	NO	Identificador de la titulación
	Ds_titulacion	NO	NO	Nombre de la titulación
	Créditos_ titulacion	NO	NO	Número total de créditos ECTS de la titulación
	Num_max_conv	NO	NO	Número máximo de convocatorias que tiene un alumno para aprobar cualquier asignatura de la titulación
	Fecha_imp	NO	NO	Concatenación del año y mes de implantación de la titulación
	Fecha_fin	NO	NO	Concatenación del último año y mes en el que se imparte la titulación y cualquiera de sus asignaturas. Si toma valor 999912 significa que la titulación aún se imparte.

UNIVERSIDAD	Id_universidad	SI	NO	Identificador de la universidad
	Ds_universidad	NO	NO	Nombre de la universidad

Tabla 2. Tablas de entidades del esquema E/R

A*: Identificador alternativo.

Las relaciones establecidas en el modelo:

RELACIÓN Y CARDINALIDAD	ATRIBUTO	IDENTIFICADOR	OPCIONAL	DESCRIPCIÓN
CURSA (N:M)	DNI	SI	NO	DNI del alumno que cursa una asignatura
	Id_asignatura	SI	NO	Identificador de la asignatura que cursa el alumno
	Anio_academico	NO	NO	Concatenación del primer año que conforma el año académico las dos últimas cifras del segundo año. (Ejemplo: año académico 2002-2003 = 2000203)
	Cuatrimestre	NO	NO	Cuatrimestre, primero o segundo, en que el alumno cursa la asignatura
	Turno	NO	NO	Turno, mañana o tarde, en que el alumno cursa la asignatura
	Num_ convocatoria	NO	NO	Numero de convocatoria de la asignatura, en la que se encuentra el alumno en el año académico.
	Num_dispensas	NO	NO	Número de veces que el alumno ha dispensado justificadamente la asignatura hasta ese año académico
	Num_libres_	NO	NO	Número de veces que el alumno ha dispensado

	dispensas			injustificadamente la asignatura hasta ese año académico
	Nota	NO	NO	Nota del alumno en la asignatura en el año académico
	Libre_eleccion	NO	SI	Toma valor s el alumno cursa la asignatura como libre elección
IMPARTE (N:M)	DNI	SI	NO	DNI del profesor responsable de la docencia
	Id_asignatura	SI	NO	Identificador de la asignatura impartida
	Anio_academico	NO	NO	Concatenación del primer año que conforma el año académico las dos últimas cifras del segundo año.
	Cuatrimestre	NO	NO	Cuatrimestre en el que se imparte la asignatura. Para un año solo se imparte en un cuatrimestre
	Nota_docencia	NO	NO	Nota que obtiene el profesor y evalúa su docencia en la asignatura y año académico
MATRICULA (N:M:P)	Id_centro	SI	NO	Identificador del centro en el que se matricula un alumno
	Id_titulacion	SI	NO	Identificador de la titulación en el que se matricula un alumno
	DNI	SI	NO	DNI del alumno matriculado en la titulación
	Anio_academico	NO	NO	Año académico de la matriculación en la titulación y centro por el alumno. Expresado como la concatenación del primer año que conforma el año académico las dos últimas cifras del segundo año.
	Convocatoria	NO	NO	Convocatoria, ordinaria o extraordinaria, en la que se oferta la titulación en el centro, para la que se corresponden las notas de acceso y en la que accede el alumno
	Anio_ academico_fin	NO	NO	Año académico en el que el alumno concluye la titulación. Expresado como la concatenación del primer año que conforma el año académico las dos últimas cifras del segundo año.
	Plazas	NO	NO	Número de plazas ofertadas para la titulación en

				el año académico, centro y convocatoria
	Nota_acceso_ PAU	NO	NO	Nota de acceso para alumnos cuyo tipo de acceso es “PAU”, para la titulación en el año académico, centro y convocatoria
	Nota_acceso_ FP	NO	NO	Nota de acceso para alumnos cuyo tipo de acceso es “Formación Profesional”, para la titulación en el año académico, centro y convocatoria
	Nota_acceso_ 25	NO	NO	Nota de acceso para alumnos cuyo tipo de acceso es “Mayores de 25”, para la titulación en el año académico, centro y convocatoria
	Tasa_credito	NO	NO	Tasa del crédito de las asignaturas de la titulación en el año académico que debe abonar el alumno
	Tipo_acceso	NO	NO	Tipo de acceso del alumno, prueba general, por formación superior o por el acceso a mayores de 25 años
	Nota_acceso	NO	NO	Nota con la que accede a la titulación en el centro

Tabla 3. Tablas de relaciones del esquema E/R

Supuestos semánticos implícitos (SSI)

SSI_1: El centro, que pertenece al campus que constituye una universidad, oferta titulaciones, por ende, el campus oferta titulaciones y la universidad oferta titulaciones.

SSI_2: Las titulaciones ofertan asignaturas. Las asignaturas están asignadas a departamentos. Por SSI_1, el campus y la universidad ofertan titulaciones. Así, el campus y la universidad tienen departamentos.

SSI_3: El alumno se matricula de un/as asignatura/s, cada una ofertada por una titulación. Por SSI_1, el centro, el campus y la universidad tienen alumnos matriculados.

SSI_4: El alumno tiene como profesor/es en un año académico a aquel/aquellos que imparten una asignatura en el mismo año académico y en el que el alumno cursa dicha asignatura.

SSI_5: El profesor tiene como alumno/s en un año académico a aquel/aquellos que cursan la/s asignatura/s en el mismo año académico que el profesor imparte.

Supuestos semánticos explícitos (SSE)

SSE_1: Un centro establece tres notas de acceso para cada titulación ofertada en un año académico y convocatoria. La nota de acceso del alumno debe ser igual o mayor a la nota de acceso, correspondiente al tipo de acceso del alumno, establecida en la oferta de la titulación, para que el alumno pueda matricularse.

SSE_2: Un alumno sólo puede cursar asignaturas pertenecientes a una titulación en la que se encuentre matriculado.

SSE_3: Un alumno ha aprobado una asignatura cuando su nota es igual o superior a 5,00.

SSE_4: Un alumno completa una titulación cuando ha cursado y aprobado tantas asignaturas ofertadas por la titulación como sean necesarias para que los créditos de estas asignaturas sumen el número de créditos o más que tiene la titulación.

SSE_5: El año académico en que un alumno completa una titulación, es el máximo año académico en que el alumno se matricula y aprueba una asignatura de la titulación (habiéndose cumplido el SS_4).

SSE_6: Un alumno no puede cursar una asignatura (convocatorias) un número mayor de veces que el número máximo de convocatorias establecidas para la titulación a la que pertenece la asignatura, descontándose del cómputo aquellas convocatorias para las que existe alguna dispensa (libre o justificada)

SSE_7: Cada asignatura esta asignada a un único departamento, a su vez la asignatura se imparte por uno o varios docentes durante un determinado año académico y cuatrimestre. En un mismo año académico y cuatrimestre, un profesor sólo puede impartir asignaturas asignadas al mismo departamento. Este departamento será al que esté asignado el profesor en ese año académico y cuatrimestre.

SSE_8: Una titulación tendrá fecha fin cuando su estado sea ‘Extinguida’.

SSE_9: Una titulación podrá pasar a estado ‘Extinguida’ cuando todos los alumnos matriculados en dicha titulación la finalicen (SSE_4) o no puedan cursar más asignaturas de la titulación (SSE_6).

SSE_10: Un alumno no podrá matricularse en una titulación cuyo estado sea ‘Extinguida’.

SSE_11: Un alumno no puede cursar una asignatura si ya la ha aprobado (SSE_3).

SSE_12: Un alumno puede cursar asignaturas no pertenecientes a la titulación de la que matricula si esta asignatura se toma como ‘Libre elección’.

SSE_13: Un alumno cursa una asignatura como libre elección cuando el atributo LIBRE_ELECCION toma valor. Dicha asignatura computará en los créditos de libre elección de aquella o aquellas titulaciones.

SSE_14: La asignatura Prueba Inglés I y Prueba Inglés II no tiene límite de convocatorias.

SSE_15: Un profesor no puede impartir una asignatura en un año académico si no hay alumnos que cursen dicha asignatura en el mismo año académico.

SSE_16: El turno pertenece a un dominio cerrado y puede tomar los valores de 'MAÑANA' o 'TARDE'.

SSE_17: La convocatoria de una matrícula pertenece a un dominio cerrado y puede tomar los valores de 'ORDINARIA' o 'EXTRAORDINARIA'.

SSE_18: El cuatrimestre pertenece a un dominio cerrado y puede tomar los valores de 'PRIMERO', 'SEGUNDO' o 'EXTRAORDINARIO'.

4.1.3. Esquema relacional

Del esquema *E/R* se desprende el *esquema relacional* mostrado en la Ilustración 40. Esquema relacional ODUNI:

Ilustración 42. Esquema relacional ODUNI

Es este *esquema relacional* a partir del que se implementará la base de datos ODUNI, la especificación técnica de cada tabla, sus campos y tipo de datos, así como los scrips SQL, corresponden al capítulo 5, IMPLEMENTACIÓN DEL SISTEMA.

4.2. Conversión a Esquema en Estrella

El *esquema en estrella* presenta una serie de *tablas de dimensiones* y una (o más) de *tabla hechos*.

Cada *tabla de dimensiones* representa una "variable" con la que definir o dar significado a un *hecho* cuantificable de un modelo. Suelen contener menos de un millón de filas y bastantes columnas descriptivas, por lo que generalmente son textuales.

Son tablas altamente indexadas. Cada *tabla de dimensiones* se combina mediante clave ajena con la/s *tabla/s de hechos*.

La *tabla de hechos* se caracteriza por tener pocos atributos, lo que se traduce en pocas columnas. Tiene al menos una por cada *tabla de dimensiones* con la que se relaciona. Habitualmente, se accede a la información mediante las *dimensiones*.

Son tablas de millones o miles de millones de registros (filas), donde la clave de cada uno es la concatenación de todos los campos que la relacionan con las *dimensiones* y el resto de campos son los *atributos de hechos*, campos generalmente de tipo numérico que son resúmenes de datos, marcas de aplicación de un método o que indican la inexistencia de valor.

Las relaciones 1:N en un esquema *E/R* pasan, generalmente, a formar jerarquías de una *dimensión* en los *esquemas en estrella*. Siendo cada entidad un nivel de esa jerarquía.

Los identificadores de las entidades sirven para generar las claves primarias de los niveles de las dimensiones, generalmente llamadas claves subrogadas, que con frecuencia son enteros autonuméricos.

Los atributos cualitativos de las entidades, pasan a ser atributos del nivel de la *dimensión*, mientras que los atributos cuantitativos, pasan a formar parte de los *atributos de hechos*.

Un claro ejemplo de esta situación, es la transformación de las entidades Financiación, Universidad, Campus y Centro en una dimensión con una jerarquía de cuatro niveles:

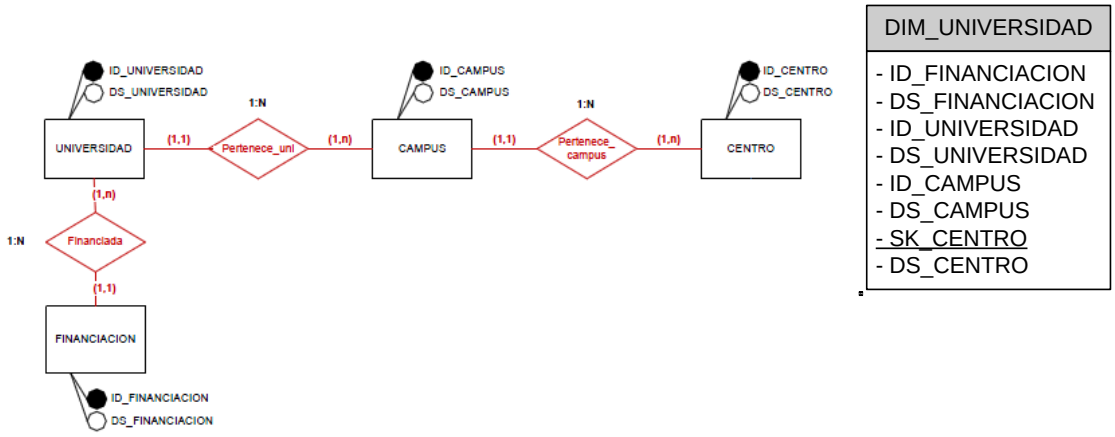


Tabla 4. Dimensión Universidad

Sin embargo, las reglas de conversión no se cumplen estrictamente. Dependiendo del contexto, del tipo de análisis o incluso por cuestiones de eficiencia de los programas las jerarquías pueden dividirse en *dimensiones* independientes o en jerarquías con caminos alternativos.

También puede ocurrir que atributos de una entidad pasen a formar *dimensiones* independientes, debido a que aportan gran significado en un análisis.

En el caso concreto que ocupa a este proyecto se razonará cada conversión del modelo.

En la imagen anterior se muestra la conversión de las entidades Financiación, Universidad, Campus y Centro en la dimensión Universidades, el nombre es lo más representativo de la dimensión.

La titulación es una entidad con muchas relaciones, siguiendo los pasos descritos debería crearse una *dimensión*, Titulación, con tres niveles de jerarquía, el mayor sería un camino alternativo entre tres opciones: estado, tipo de titulación y grupo de estudios. El nivel intermedio de la jerarquía sería la titulación y el más bajo la asignatura.

Una alternativa a una única *dimensión*, con un camino alternativo entre tres, serían tres dimensiones diferentes con los niveles intermedio y último igualmente definidos. Esta solución triplica la información de los últimos niveles, quedando descartada.

La opción elegida, para la transformación de las entidades en la *dimensión* Titulación, es intermedia. La entidad Estado forma una dimensión independiente, mientras que las entidades Tipo_titulacion y Grupo_estudio son el nivel superior de una *dimensión* (representan un camino alternativo) cuyos niveles siguientes lo forman las entidades Titulación y Asignatura, en ese orden.

El camino alternativo lo forman esas dos entidades y no otras porque tienen un significado similar y los análisis se realizan desde el punto de vista de tipo de titulación o grupo de estudios de ésta, alternativamente.

Dado que el camino alternativo no se puede definir en medio de una *dimensión*, por cuestiones técnicas, el estado no puede establecerse en la jerarquía como un primer nivel (sobre el tipo de titulación y grupos de estudios), por lo que conforma una *dimensión* independiente.

Los créditos de la asignatura, como atributo cuantitativo, formarán parte de las *medidas*. Por otro lado, el atributo curso de la entidad Asignatura aporta mucho valor en un análisis, por lo que pasa a formar una *dimensión*.

En este punto surge una alternativa, el curso puede ser parte de la jerarquía en la *dimensión* titulación, una asignatura es de un sólo curso, luego podría descender de este nivel.

Sin embargo, si se toma esta decisión no se pueden analizar otras *dimensiones* conjuntamente con el curso. Es decir, si curso no es una *dimensión* separada no se puede analizar la nota media (*medida*), por ejemplo, de un campus (*dimensión*) en un curso, dado que para acceder al curso habría que seleccionarlo en todas las titulaciones (nivel inmediatamente superior al curso) del campus.

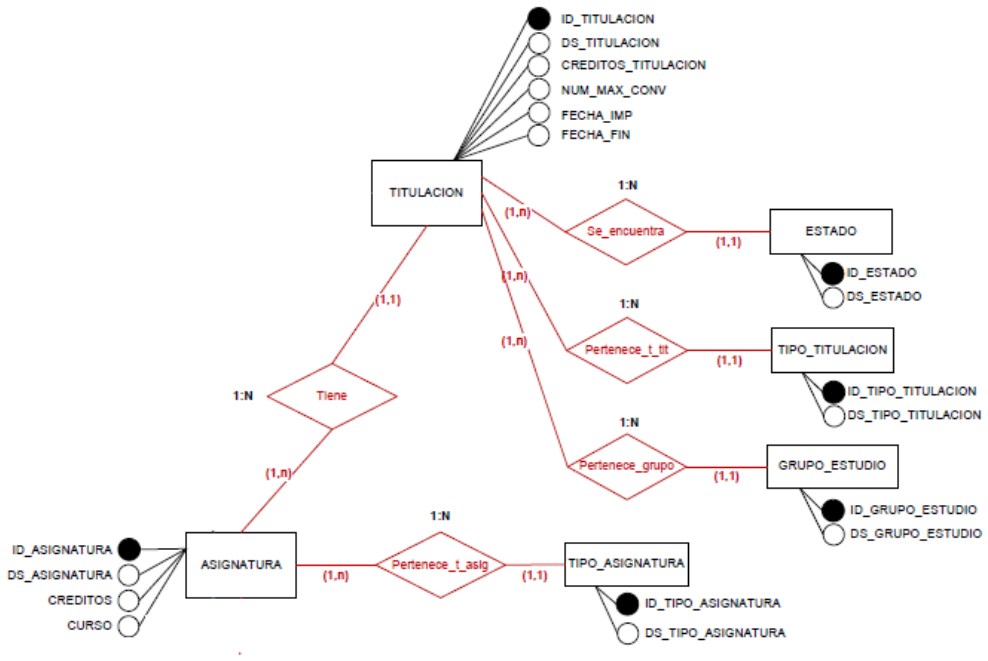


Tabla 5. Dimensiones Titulación, Curso y Estado

La entidad Asignatura, que forma parte de una *dimensión* establecida estable dos relaciones, con las entidades Tipo_Asignatura y Departamento.

El motivo por el que no son parte de la *dimensión* en la que se encuentra asignatura, como niveles superiores de asignatura es análogo al de curso. Igualmente, es el motivo por el que ambas se establecen como dimensiones separadas.

De esta forma se pueden analizar conjuntamente con otras *dimensiones* el tipo de asignatura y los departamentos.

En el esquema *E/R*, la entidad Departamento se relaciona 1:N con Categoría_departamento, se forma una *dimensión* jerarquizada.

En esta parte no aparecen atributos de hechos.

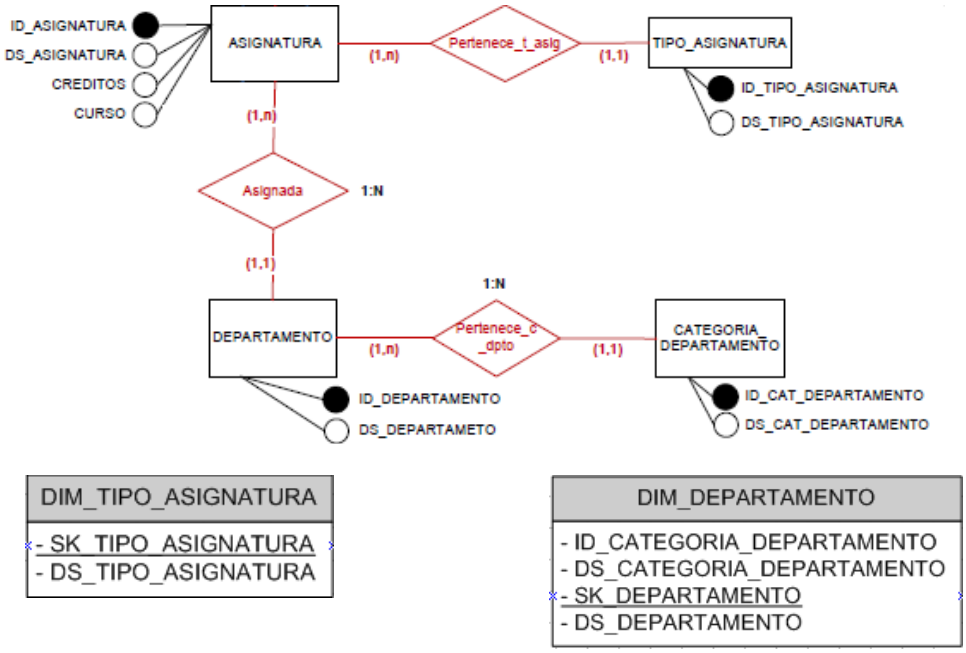


Tabla 6. Dimensiones Tipo de asignatura y Departamento

Las entidades Alumno y Docente son hijas de Persona, en el *esquema en estrella* pasan a formar una *dimensión* cada una, los atributos cualitativos (nombre y apellidos) de Persona son heredados y pasan como atributos de la dimensión. Estas *dimensiones* se llamarán Maestro de alumnos y Maestro de docentes, por ser un registro único de estos tipos de personas.

El atributo anio_nacimiento, como cuantitativo, pasará a los *atributos de hechos*, para con él, pre-calcular la edad de cada individuo.

El atributo cualitativo Sexo, puede resultar de utilidad como *dimensión* de algunos análisis. Dado que hace referencia al sexo de dos tipos de personas, y cada una establece una dimensión, el atributo origina dos *dimensiones* independientes.

La entidad Categoría_laboral forma otra *dimensión*, y su único atributo cuantitativo, Tasa_docente_E. Este atributo no pasa al modelo del *cubo* porque resulta equivalente a la categoría laboral.

Ni el tipo de persona, ni la categoría laboral, ni el sexo, forman parte de una jerarquía en las dimensiones de alumno y docente. Esto ocurre en este modelo concreto porque no se pretende hacer ningún análisis focalizando en un alumno o profesor concreto, por lo que no tienen sentido como dimensión, pero son necesarios para establecer el número de alumnos o profesores y otros cálculos que precisan de esta cifra.

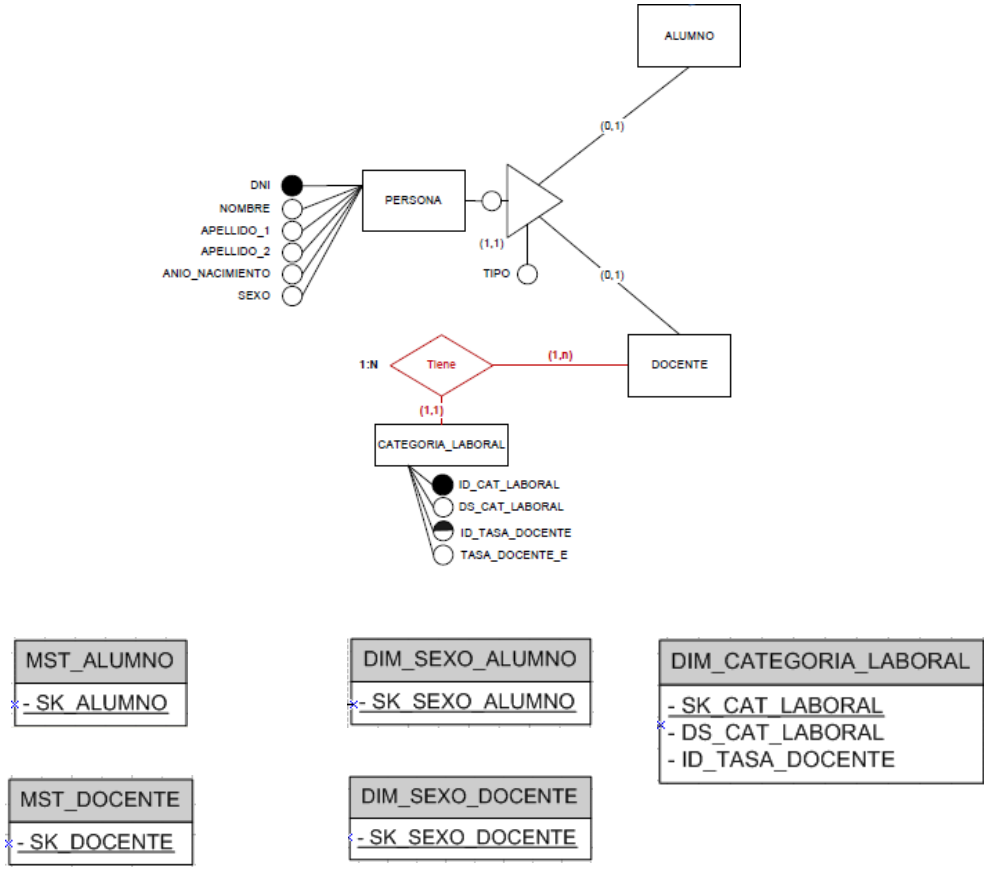


Tabla 7. Dimensiones Alumno, Docente, Sexo y Categorial laboral

De las relaciones N:M y N:M:P del modelo se extraen la mayoría de las *medidas*, los atributos de estas relaciones son en su mayoría cuantitativos, y de aquellos cualitativos se establecen nuevas dimensiones. En estas relaciones suelen crearse atributos de tiempo, en este caso con ellos se definirá la dimensión temporal, casi indispensable en modelos multidimensionales, donde los análisis suelen comparar información a lo largo del tiempo. Estas relaciones enlazan el resto de entidades que ya han conformado *dimensiones*.

El año académico, es común en las tres relaciones con atributos, formará la *dimensión* tiempo junto con cuatrimestre, como subnivel.

La convocatoria es una *dimensión* de análisis especialmente interesante para las notas obtenidas por los alumnos. El atributo Num_comvocatoria de la relación Cursa hace referencia a la convocatoria del examen de la asignatura, mientras que Convocatoria de la relación Matricula referencia la convocatoria de acceso a la universidad.

El tipo de acceso a la universidad (Matricula), y turno (Cursa) son *dimensiones* separadas, dado que aportan significado a un análisis.

Los atributos Tasa_credito, Nota_acceso (relación Matricula), Num_dispensas, Num_libres_dispensas, Nota (relación Cursa) y Nota_docencia (relación imparte) son *medidas*.

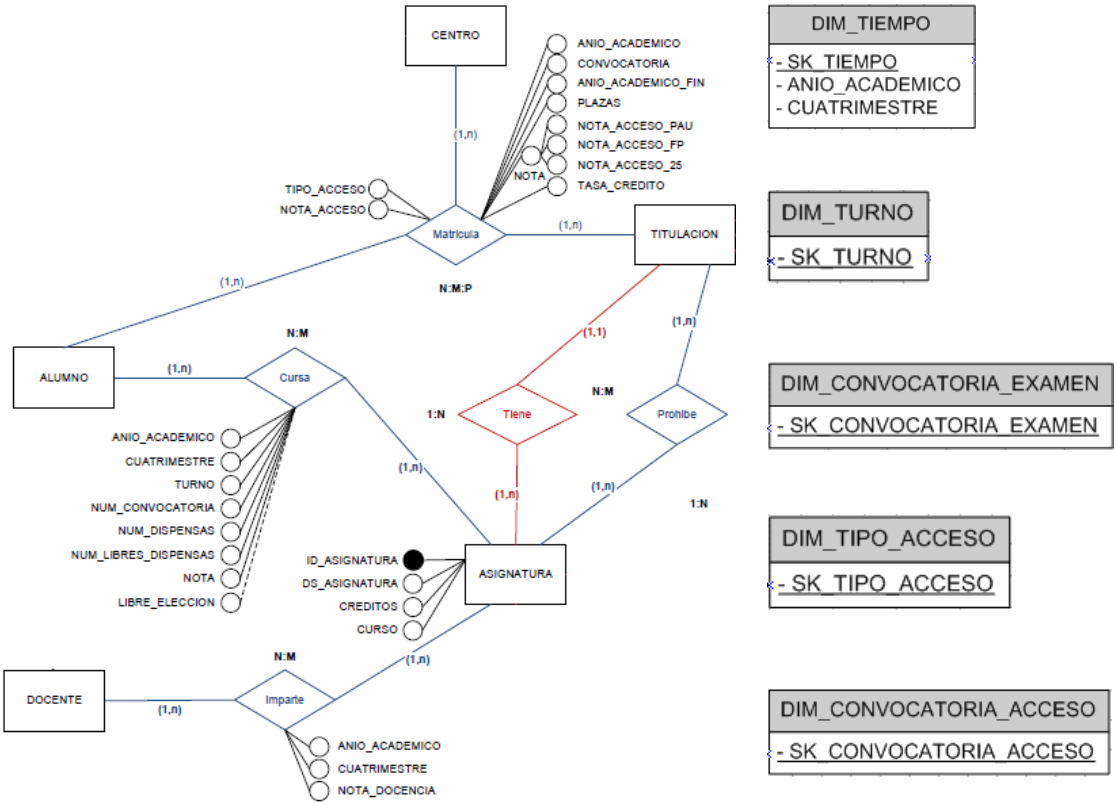


Tabla 8. Dimensiones Tiempo, Turno, Convocatoria, Tipo de acceso y Convocatoria de acceso

Para definir la *tabla de hechos* se precisan las claves de las dimensiones con que se relacionan y los *atributos de hecho*: anio_nacimiento, Tasa_credito, Nota_acceso, Num_dispensas, Num_libres_dispensas, Nota, Nota_docencia.

Al existir atributos que son propios de los profesores, y que no pueden repetirse por cada alumno, como la Nota_docencia y la edad_docente, se definen dos *tablas de hechos.*

El *esquema en estrella* final queda diseñado:

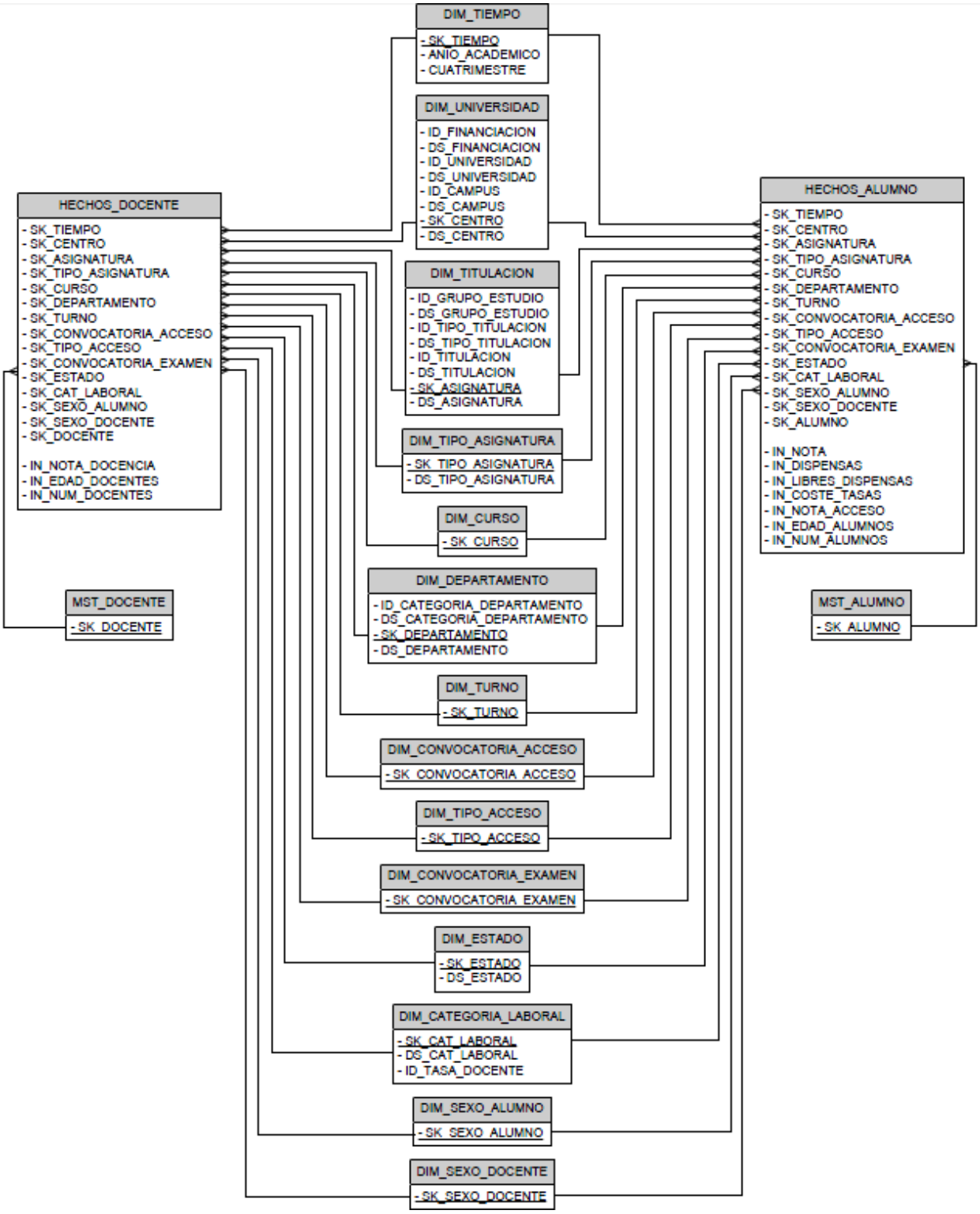


Ilustración 43. Esquema en estrella ODUNI

4.3. Conversión a Esquema en Copo de Nieve

El *esquema en copo de nieve* es una variante del *esquema en estrella* que presenta las *tablas de dimensiones* en más de un nivel, normalizándolas.

Para definir la conversión se partirá del *esquema en estrella*.

Las dimensiones con más de un nivel se normalizan y dividen en diferentes tablas de dimensiones. Este es el caso de las tablas DIM_TIEMPO, DIM_UNIVERSIDAD, DIM_TITULACION, DIM_DEPARTAMENTO.

Las tablas que se desarrollan de una dimensión jerarquizada del *esquema en estrella* se enlazan, de forma que la tabla correspondiente a la jerarquía menos arrastra la clave ajena de su nivel superior, así hasta completar la *dimensión*.

El esquema desarrollado es como sigue:

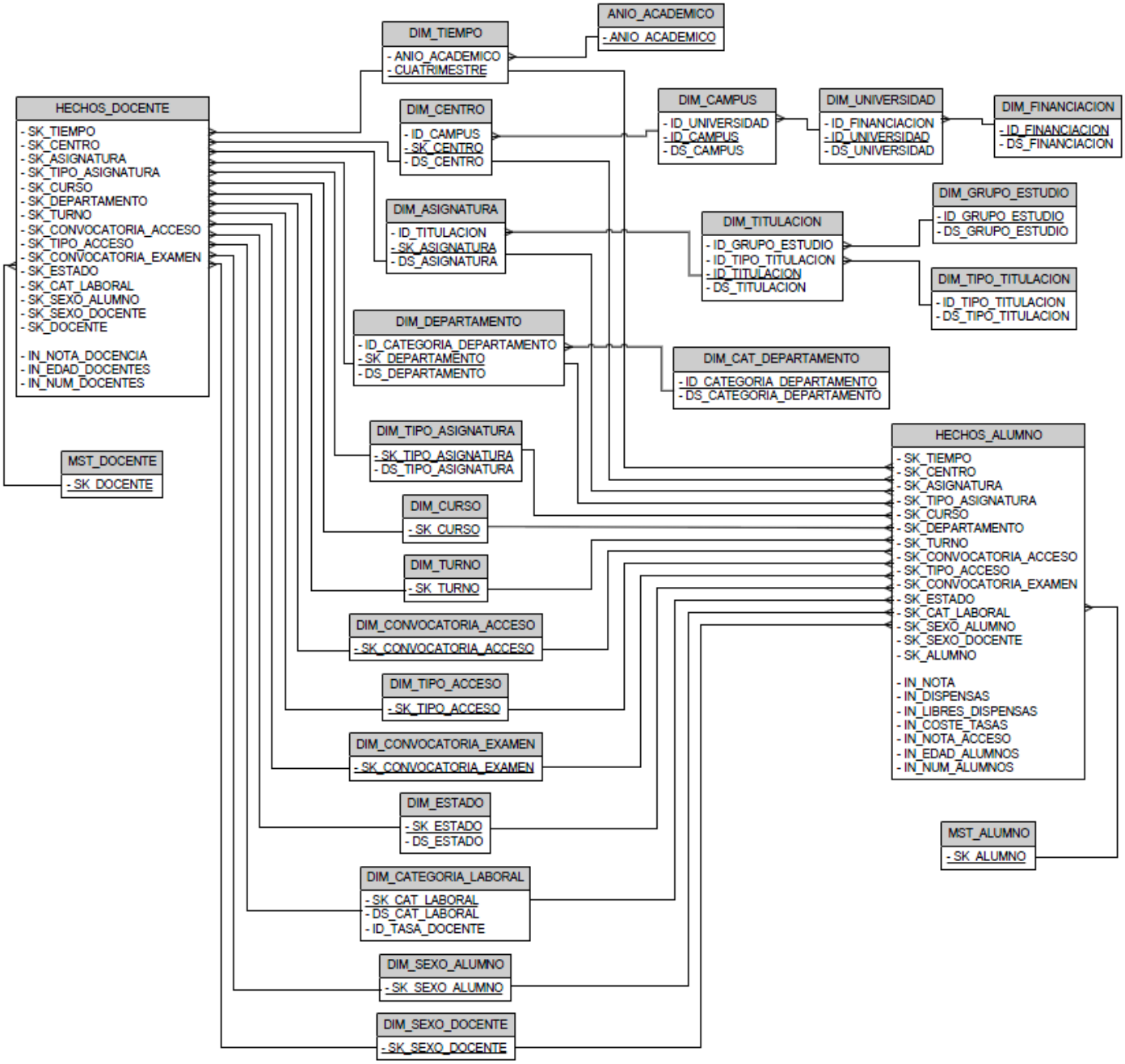


Ilustración 44. Esquema en copo de nieve

4.4. Estudio de la automatización del proceso de conversión entre modelos

Una automatización del proceso de conversión perfecta debería tener en cuenta la semántica del modelo, no reflejada formalmente en un esquema relacional. Sin tener en cuenta este aspecto podría llevarse a cabo una conversión entre modelos relacionales más o menos válida de forma automática.

Partiendo de un esquema relacional se llegaría a una aproximación de un esquema en estrella o en copo de nieve de la forma siguiente:

1. Para cada tabla del modelo relacional se genera una tabla, donde los campos son los atributos cualitativos y la clave primaria el atributo identificativo.

2. Las relaciones 1:N permiten extender las claves entre las tablas que se relacionan, estas tablas originan una única tabla donde la clave primaria es la clave de la tabla del extremo N de la relación.

3. En los modelos en estrella, estas tablas pueden encadenar relaciones, originando tablas de dimensiones jerarquizadas, donde el extremo 1 de la relación es el nivel superior del extremo N. En los modelos en copo de nieve las tablas no se agrupan en una única.

4. Los atributos cuantitativos originarán una nueva tabla, siendo cada uno de ellos una columna. En esta misma tabla, se añaden tantas columnas como identificadores tengan las tablas de donde se extraen los atributos cuantitativos. Esta conformará la tabla de hechos del esquema.

5. Si la tabla de hechos contiene claves identificativas que no aparecen en ninguna otra tabla definida se crea una tabla, cuya columna sea este campo identificativo.

4.5. Cubo multidimensional UNICUBE

El diseño del *cubo* se basa en el *esquema en estrella*. En él se definen las mismas *dimensiones* y *tablas de hechos*, y es la aplicación las que las relaciona (en función del nombre del campo) y ofrece lo datos calculados para un análisis on-line, cómodo y con un formato amigable.

El *cubo*, definido con el componente de Cognos Transformer, no obtiene la información directamente de la BD, sino que la extrae de un modelo de datos generado con el componente de Cognos Framework Manager y publicado en Cognos Connection.

El modelo de datos publicado, llamado Modelo UNICUBE, es un *framework* que supone una capa intermedia entre el origen puro de datos, en un *esquema en estrella*, de la BD ODUNI y el cubo UNICUBE.

Un *framework* o *infraestructura digital*, es una estructura conceptual y técnica definida, con base a la cual un proyecto de *software* puede ser más fácilmente organizado y desarrollado.

Los *cubos* pueden leer de diferentes orígenes, sin ser necesario un *framework* como el generado, sin embargo, la creación de Modelo UNICUBE ofrece la posibilidad de validar la información, añadir campos adicionales para descripciones o códigos de ordenación, permite la generación de campos calculados nuevos en base a medidas almacenadas en el modelo (gracias a los operadores definidos en Framework Manager), etc.

Además, la BD puede contener tablas, resultado de pasos intermedios de la *ETL*, o campos en las tablas, que no sean necesarios en el *cubo*. A través de un *framework*, se puede cribar la información que alimenta el *cubo*.

Un *cubo* consta de *modelo* y *cubo* como entidad con datos e información. El *modelo* es la definición de la estructura del cubo, sus *dimensiones*, *medidas* y parámetros que lo configuran (paquete de datos que lee, carpeta de publicación…). Un *cubo* es la imagen de los datos una vez se genera el *modelo*. Es decir, los análisis de un *cubo* son datos calculados y almacenados en la BD en un momento del tiempo, si estos datos cambian en el origen no cambian en el *cubo* hasta que este se genera de nuevo.

4.5.1. Framework "Modelo UNICUBE"

Para que Framework Manager pueda leer la información contenida en ODUNI es necesario crear una base de datos ODBC, ya que el componente no puede establecer una conexión directamente con el motor SQL Server 2000. Windows XP disponen de un componente que facilita estas operaciones (véase Anexo B).

Una vez el programa Framework establece la conexión con la base de datos ODUNI se pueden generar las consultas sobre las tablas del *esquema en estrella*. Las consultas originan tablas en el *framework*. Para estructurar la información, pensando en facilitar evoluciones futuras del proyecto, en el *framework* se establecen dos espacios:

- **Capa de BBDD:** se trata de una capa del *framework* que copia íntegramente las tablas de la BD que serán necesarias para el *cubo*, generalmente son las *tablas de dimensiones* y *tablas de hechos*. En ocasiones, se añaden a este espacio tablas de datos agregados útiles para realizar informes.

- **Capa de Transformer:** espacio o capa de donde leerá la información el *cubo*, se trata de una capa donde las tablas definidas se basan en la información de la

Capa de BBDD, no leyéndose directamente de la BD. Estas tablas definen campos nuevos que pueden ser útiles para el *cubo*, como datos pre-calculados o campos auxiliares.

El programa permite generar *paquetes* con visibilidad parcial de los elementos generados en el *framework*. La Capa de BBDD ya supone un filtro que elimina las tablas de la BD que no son de interés. Modelo UNICUBE sólo publica el espacio Capa de Transformer, con lo que el cubo solo tendrá acceso a la información útil de las tablas de interés.

Las tablas que son declaradas en la Capa de BBDD son las 16 dimensiones establecidas en el *esquema en estrella* y las dos *tablas de hechos*.

Las tablas definidas en la Capa de Transformer son estas mismas pero algunas de ellas presentan cambios en algunos campos descriptivos. Por ejemplo, la *dimensión* tiempo, almacena el año académico como el concatenado del primer año y las dos últimas cifras del segundo. Mostrar en el cubo 200340 hace poco comprensible la lectura al usuario. En la capa de Transformer, haciendo uso de los operadores y funciones definidos en Framework Manager, se separa este campo y se introduce una barra "/", de modo que al *cubo* llegue el literal 2003/04, que si entiende el usuario. Realizar estas operaciones en el *framework* libera al motor de base de datos de operaciones que tienen una finalidad de formato, ya que la información es la misma.

4.5.2. Dimensiones UNICUBE

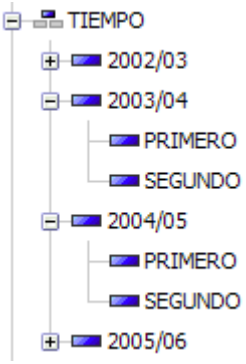
Las *dimensiones* son atributos relativos a las *medidas*, según las cuales se da la perspectiva de análisis a estas.

Una *dimensión* representa una variable de análisis del negocio. Tiene una estructura jerárquica que facilita el estudio de los *indicadores* del modelo. Puede contener uno o varios niveles. Estos niveles, a su vez, pueden estar formados por uno o varios elementos.

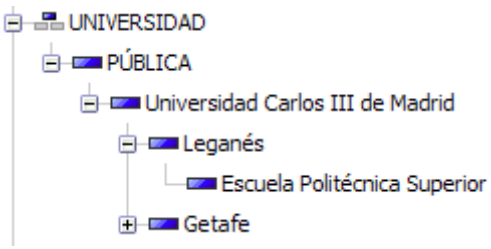
Ilustración 45. Dimensión Tiempo

A continuación, las *dimensiones* definidas para UNICUBE y una visión de estas en un cubo generado:

4.5.2.1. Tiempo

VISTA EN EL CUBO	DESCRIPCIÓN
	Para el análisis de la información se recogerán datos discretos cuatrimestre de cada año académico, permitiendo así obtener en esta dimensión datos tanto a nivel cuatrimestral como acumulado por periodos anuales. La *dimensión* de tiempo incluye los siguientes niveles: • ***Año académico:*** Comprende el periodo de tiempo desde Septiembre de un año hasta el del año siguiente. • ***Cuatrimestre:*** Corresponde a los cuatrimestres educativos estipulados. Mediante esta dimensión es posible hacer que la medida visualizada sea neta o acumulada. Teniendo varios casos: ▪ Si no se aplica ningún filtro temporal, se ven los datos acumulados de toda la historia registrada. ▪ Si se aplica un filtro por un año académico, se ve el acumulado de todo ese año. ▪ Un filtro por cuatrimestre, mostrará el acumulado de este

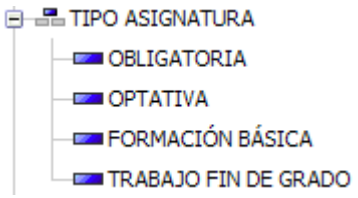
4.5.2.2. Universidad

VISTA EN EL CUBO	DESCRIPCIÓN
	Representa la universidad objeto de análisis. La dimensión de mercado incluye los siguientes niveles: • ***Tipo de financiación:*** Incluye las universidades agrupadas según sean Públicas o Privadas. • ***Universidad:*** Universidades incluidas en el estudio. • ***Campus:*** Las universidades se componen de campus, son generalmente partes de la universidad distribuidas en lugares geográficamente distintos. • ***Centro:*** Para un mismo campus podemos encontrar diferentes centros de estudios, relacionados generalmente con el tipo de titulaciones que oferta.

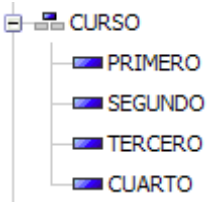
4.5.2.3. Titulación

VISTA EN EL CUBO	DESCRIPCIÓN
Vista hasta el nivel de titulación TITULACION TITULACION TIPO DE TITULACION GRADO Grado en Humanidades (50+) Grado en Ingeniería Eléctrica Grado en Ingeniería en Tecnologías Industriales GRUPO DE ESTUDIOS Humanidades Grado en Humanidades (50+) Ingeniería Grado en Ingeniería Eléctrica Grado en Ingeniería en Tecnologías Industriales	Representa las titulaciones que se analizan. Esta dimensión presenta un camino alternativo para su análisis en el primer nivel. El principal, desde el punto de vista del tipo de titulación (grado, ingeniería, licenciatura…) y uno secundario, por el grupo de estudios (Humanidades, ingeniería…) Los niveles siguientes son idénticos en los dos caminos de análisis y son: • ***Titulación*** • ***Asignatura***
Vista hasta el nivel de asignatura por el camino Tipo de Titulación TITULACION TITULACION TIPO DE TITULACION GRADO Grado en Humanidades (20 de 50+) Prehistoria e Historia Antigua Historia del Arte I Movimientos Literarios Lengua Española Cultura Clasica Historia Medieval Historia de la Filosofia I Geografia Tecnicas de busqueda y uso de la informacic Tecnicas de expresion oral y escrita Historia Moderna Teoria de la Literatura Historia de la Filosofia II Lingüistica General Geografia Regional Historia Contemporanea Historia del Arte II Literatura Comparada Filosofia Moral Semantica y Pragmatica del Español Más Grado en Ingeniería Eléctrica	

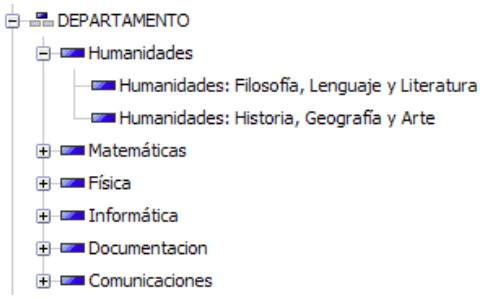
4.5.2.4. Tipo asignatura

VISTA EN EL CUBO	DESCRIPCIÓN
	Con esta dimensión se establecen los diferentes tipos de asignaturas que puede tener una titulación. La dimensión incluye un nivel: • ***Tipo asignatura***

4.5.2.5. Curso

VISTA EN EL CUBO	DESCRIPCIÓN
	En la dimensión Curso figuran los diferentes cursos que puede tener una titulación. La dimensión de Curso incluye un único nivel: • ***Curso***

4.5.2.6. Departamento

VISTA EN EL CUBO	DESCRIPCIÓN
	El nivel inferior representa los diferentes departamentos en los que se estructura la enseñanza en una universidad. Existen dos niveles en esta dimensión: • ***Categoría departamental*** • ***Departamento*** Un departamento tiene a su cargo una serie de titulaciones, y a su vez, una titulación suele depender de varios departamentos. Dado que las universidades son independientes no tienen porque componerse de los mismos departamentos y estos no tienen por qué tener a su cargo las mismas titulaciones, por este motivo se establece el nivel superior Categoría departamental, que trata de crear equivalencias entre diferentes departamentos con el fin de poder comparar diferentes universidades con esta dimensión.

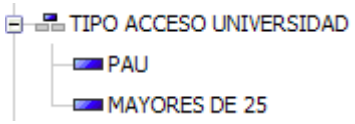
4.5.2.7. Turno

VISTA EN EL CUBO	DESCRIPCIÓN
	En la dimensión Turno se dividen los posibles turnos en los que se imparte la docencia a los alumnos, mañana o tarde. La dimensión de Turno incluye un único nivel: • ***Turno***.

4.5.2.8. Convocatoria

VISTA EN EL CUBO	DESCRIPCIÓN
	En la dimensión Convocatoria se dividen las veces que un alumno acude al examen de la misma asignatura. La dimensión incluye un único nivel: • ***Convocatoria***.

4.5.2.9. Tipo acceso universidad

VISTA EN EL CUBO	DESCRIPCIÓN
	Representa el camino del alumno para acceder a la universidad. La dimensión de Tipo acceso universidad se compone del único nivel: • ***Tipo acceso universidad*** Los tipos de acceso definidos son: • **PAU:** Prueba general de acceso, a través de Bachiller • **Mayores de 25:** el acceso a la universidad teniendo 25 años o más. • **FP:** Acceso a través de módulos de Formación Profesional.

4.5.2.10. Convocatoria acceso

VISTA EN EL CUBO	DESCRIPCIÓN
CONVOCATORIA ACCESO ORDINARIA EXTRAORDINARIA	Diferencia la convocatoria en la que el alumno supera las pruebas de acceso a la universidad. Esta dimensión incluye un único nivel: • ***Convocatoria acceso*** Los valores que puede tomar el nivel son: • **Ordinaria:** primera convocatoria, suele corresponder al mes de junio**.** • **Extraordinaria:** segunda convocatoria, generalmente en septiembre.

4.5.2.11. Categoría laboral

VISTA EN EL CUBO	DESCRIPCIÓN
CATEGORÍA LABORAL CATEDRÁTICO DOCTOR AYUDANTE DOCTOR TITULAR ASOCIADO	Condición profesional que tiene un profesor en la universidad donde imparte una asignatura. Esta categoría está asociada a la tasa. La dimensión Categoría laboral tiene el nivel: • ***Categoría laboral*** Ésta se organiza de la siguiente manera: • ***Catedrático*** • ***Doctor*** • ***Ayudante Doctor*** • ***Titular*** • ***Asociado***

4.5.2.12. Sexo alumno

VISTA EN EL CUBO	DESCRIPCIÓN
SEXO ALUMNO MUJER VARON	Representa el sexo, mujer o varón, del alumno matriculado en una asignatura. La dimensión incluye un único nivel: • ***Sexo alumno***

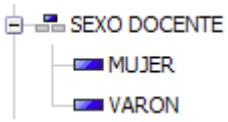
4.5.2.13. Sexo docente

VISTA EN EL CUBO	DESCRIPCIÓN
SEXO DOCENTE MUJER VARON	Representa el género, masculino o femenino del docente que imparte la asignatura. La dimensión incluye un único nivel: • ***Sexo docente***

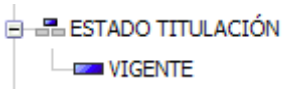
4.5.2.14. Estado titulación

VISTA EN EL CUBO	DESCRIPCIÓN
ESTADO TITULACIÓN VIGENTE	La dimensión Estado Titulación permitirá obtener la información referenciada al estado de una titulación respecto al plan educativo. La dimensión de Estado incluye un único nivel: • ***Estado titulación*** El estado de una titulación puede ser: • **Vigente:** la titulación forma parte del plan educativo actual. • **En extinción:** el plan no la incluye la titulación pero aún se imparte. • **Extinguida:** no se imparte la titulación.

4.5.2.15. Alumno

VISTA EN EL CUBO	DESCRIPCIÓN
	Representa cada alumno para el que existe información. La dimensión incluye un único nivel: • ***Alumno***

4.5.2.16. Docente

VISTA EN EL CUBO	DESCRIPCIÓN
	Representa cada profesor para el que existe información. La dimensión incluye un único nivel: • ***Docente***

4.5.3. Atributos de hechos, Indicadores o Medidas

Los indicadores o medidas son los datos cuantificables que están siendo analizados por las dimensiones.

El cruce de estos indicadores con las distintas dimensiones nos ayuda a localizar la causa de su valor en niveles distintos a los deseados.

Los indicadores obtenidos servirán para resaltar la evolución de las notas obtenidas tanto por alumnos como profesores, los costes de la enseñanza o la evolución de las edades de los alumnos a través del tiempo.

En el cubo multidimensional UNICUBE existen:

- Datos de análisis: Son las medidas definidas para el análisis, a partir de los atributos de hechos.

- Datos Ocultos: Son datos que no son el resultado de ningún cálculo y a partir de los cuales, se calculan los Datos de análisis.

VISTA EN EL CUBO	DESCRIPCIÓN
DATOS DE ANÁLISIS	
No. Alumnos	Representa el número de alumnos diferentes
Nota Media Acceso a la Universidad	Nota media de acceso a la universidad que obtienen los alumnos
Nota Media	Nota media de los alumnos obtenida en los exámenes
Edad Media Alumno	Edad media del alumno
Media de Dispensas	Es la media de veces que dispensan los alumnos un examen
Costes Tasas	Coste en Euros de las tasas que debe abonar el alumno a la universidad. Es el importe no financiado en el caso de las universidades públicas
No. Docentes	Representa el número de docentes
Nota Media Docente	Es la nota media obtenida por los profesores
Edad Media Docente	Edad media de los profesores
DATOS OCULTOS	
IN_NOTA	Nota del alumno, se utiliza para calcular la media
IN_NOTA_ACCESO	Nota de acceso del alumno, se utiliza para calcular la media
IN_DISPENSAS	Número de dispensas justificadas del alumno, se utiliza para calcular la media de dispensas globales
IN_LIBRES_DISPENSAS	Número de dispensas injustificadas del alumno, se utiliza para calcular la media de dispensas globales
IN_EDAD_ALUMNO	Edad del alumno, utilizada para el cálculo de la media
IN_NUM_ALUMNOS	Número de alumnos contando repetidos que existen en un cruce de dimensiones
IN_NOTA_DOCENTE	Nota de evaluación de la docencia para un profesor, utilizada para la media
IN_EDAD_DOCENTE	Edad del docente, utilizada para el cálculo de la media
IN_NUM_DOCENTES	Número de docentes contando repetidos que existen en un cruce de dimensiones

Capítulo 5

En este capítulo se describirá cómo se ha implementado el sistema analizado y diseñado en el capítulo anterior.

Se comenzará por el esquema *E/R*, que se estableció para ODUNI, y pasando por la transformación al esquema en estrella se llegará a la implementación del *cubo multidimensional* UNICUBE.

5. IMPLEMENTACIÓN DEL SISTEMA

El entorno de trabajo será Windows XP Profesional Versión 2002 Service Pack 3. Sobre este sistema operativo se implementará una base de datos relacional bajo un modelo E/R en un SQL Server 2000. Se convertirá la información de este modelo a un esquema en estrella alojado en la misma base de datos, a través de la creación de vistas generadas en SQL en el mismo motor de base de datos, en lugar de con la herramienta PowerCenter, para la que no se dispone de licencias.

Se implementará, también en este sistema operativo, un cubo OLAP con el paquete de programas IBM Cognos 8, que explote el esquema en estrella.

5.1. ODUNI

Para la creación de la base de datos, denominada ODUNI, se utiliza SQL Server 2000, la interacción con dicha BD se realiza a través de Query Analizer.

El uso que se da a la BD queda limitado a la introducción de los datos necesarios para evolucionar el caso práctico, la creación de las vistas para la conversión de un esquema *E/R* en un *esquema en estrella* y para la consulta de los datos con el fin de realizar pruebas que confirmen el correcto funcionamiento del proceso de conversión establecido entre modelos.

El usuario será una persona conocedora de la estructura interna del sistema, capaz de introducir datos correctamente y de proceder a su corrección en los casos necesarios.

5.1.1. Creación de base de datos y tablas del esquema E/R

Para implementar la BD ODUNI, se utilizó la siguiente sentencia:

Sentencia SQL para la creación de la base de datos
Create database ODUNI;

Seguidamente, las tablas creadas del esquema *E/R*, para configurar ODUNI, por orden alfabético y sus scripts en SQL.

TABLA	ALUMNO		
NOMBRE DEL CAMPO[2]	TIPO DE DATOS[3]	ADMITE NULOS	DESCRIPCIÓN
DNI (PK) (FK)	Alfanumérico (9)	NO	DNI del alumno

```
CREATE TABLE ALUMNO(
DNI                    varchar(9) PRIMARY KEY,
FOREIGN KEY (DNI) REFERENCES PERSONA
        ON DELETE CASCADE);
```

Tabla 9. Tabla Alumno

TABLA	ASIGNATURA		
NOMBRE DEL CAMPO	TIPO DE DATOS	ADMITE NULOS	DESCRIPCIÓN
Id_asignatura (PK)	Numérico	NO	Identificador de la asignatura
Ds_asignatura	Alfanumérico (100)	NO	Nombre de la asignatura
Id_titulacion (FK)	Numérico	NO	Identificador de la titulación a la que pertenece la asignatura
Id_tipo_asignatura (FK)	Numérico	NO	Identificador del tipo de asignatura de que se trata

[2] Los campos acompañados del literal *(PK)* indican que forman parte de la clave primaria de la tabla (Primary Key). Los acompañados del literal *(FK)* indican que son clave ajena de la tabla (Foreing Key). Puede ser ambas cosas simultánemaente.

[3] Tipos de datos:
Alfanuméricos, se describen como *Alfanumérico (L)*, donde L es el número máximo de caracteres almacenables.
Numéricos, se describen como *Decimal (E,D)*, donde E representa el número de dígitos de la parte entera y D el número de dígitos de la parte decimal.

Id_departamento (FK)	Numérico	NO	Identificador del departamento al que pertenece la asignatura
Creditos	Numérico	NO	Créditos ECTS de la asignatura
Curso	Numérico	NO	Curso en que se imparte la asignatura

```
CREATE TABLE ASIGNATURA(
ID_ASIGNATURA          int PRIMARY KEY,
DS_ASIGNATURA          varchar(100) NOT NULL,
ID_TITULACION          int NOT NULL,
ID_TIPO_ASIGNATURA     int NOT NULL,
ID_DEPARTAMENTO        int NOT NULL,
CREDITOS               int NOT NULL,
CURSO                  int NOT NULL,
FOREIGN KEY (ID_TITULACION) REFERENCES TITULACION,
FOREIGN KEY (ID_TIPO_ASIGNATURA) REFERENCES TIPO_ASIGNATURA,
FOREIGN KEY (ID_DEPARTAMENTO) REFERENCES DEPARTAMENTO);
```

Tabla 10. Tabla Asignatura

TABLA	CAMPUS		
NOMBRE DEL CAMPO	**TIPO DE DATOS**	**ADMITE NULOS**	**DESCRIPCIÓN**
Id_campus (PK)	Numérico	NO	Identificador del campus
Ds_campus	Alfanumérico (100)	NO	Nombre del campus
Id_universidad (FK)	Numérico	NO	Identificador de la universidad a la que pertenece el campus

```
CREATE TABLE CAMPUS(
ID_CAMPUS              int PRIMARY KEY,
DS_CAMPUS              varchar(100) NOT NULL,
ID_UNIVERSIDAD         int NOT NULL,
FOREIGN KEY (ID_UNIVERSIDAD) REFERENCES UNIVERSIDAD);
```

Tabla 11. Tabla Campus

TABLA	CATEGORIA_LABORAL		
NOMBRE DEL CAMPO	**TIPO DE DATOS**	**ADMITE NULOS**	**DESCRIPCIÓN**
Id_cat_laboral (PK)	Numérico	NO	Identificador de la categoría laboral a la que puede pertenecer un profesor
Ds_cat_laboral	Alfanumérico (15)	NO	Descripción de la categoría laboral
Id_tasa_docente	Alfanumérico (2)	NO	Identificador de la tasa correspondiente a la categoría laboral
Tasa_docente_E	Decimal (4,2)	NO	Coste en Euros de la tasa

```
CREATE TABLE CATEGORIA_LABORAL(
ID_CAT_LABORAL        int PRIMARY KEY,
DS_CAT_LABORAL        varchar(15) NOT NULL,
ID_TASA_DOCENTE       varchar(2) UNIQUE,
TASA_DOCENTE_E        decimal(4,2) NOT NULL);
```

Tabla 12. Tabla Categoria laboral

TABLA	CATEGORIA_DEPARTAMENTO		
NOMBRE DEL CAMPO	**TIPO DE DATOS**	**ADMITE NULOS**	**DESCRIPCIÓN**
Id_cat_departamento (PK)	Numérico	NO	Identificador de la categoría del departamento a la que puede pertenecer un departamento
Ds_cat_departamento	Alfanumérico (70)	NO	Descripción de la categoría del departamento

```
CREATE TABLE CATEGORIA_DEPARTAMENTO(
ID_CAT_DEPARTAMENTO          int PRIMARY KEY,
DS_CAT_DEPARTAMENTO          varchar(70) NOT NULL);
```

Tabla 13. Tabla Categoría departamento

TABLA	CENTRO		
NOMBRE DEL CAMPO	**TIPO DE DATOS**	**ADMITE NULOS**	**DESCRIPCIÓN**
Id_centro (PK)	Numérico	NO	Identificador del centro de una universidad
Ds_centro	Alfanumérico (100)	NO	Nombre del centro
Id_campus (FK)	Numérico	NO	Identificador del campus al que pertenece el centro

```
CREATE TABLE CENTRO(
ID_CENTRO              int PRIMARY KEY,
DS_CENTRO              varchar(100) NOT NULL,
ID_CAMPUS              int NOT NULL,
FOREIGN KEY (ID_CAMPUS) REFERENCES CAMPUS);
```

Tabla 14. Tabla Centro

TABLA	CURSA		
NOMBRE DEL CAMPO	**TIPO DE DATOS**	**ADMITE NULOS**	**DESCRIPCIÓN**
DNI (PK) (FK)	Alfanumérico (9)	NO	DNI del alumno que cursa una asignatura
Id_asignatura (PK) (FK)	Numérico	NO	Identificador de la asignatura que cursa el alumno
Anio_academico (PK)	Numérico	NO	Concatenación del primer año que conforma el año académico las dos últimas cifras del segundo año. (Ejemplo: año académico 2002-2003 = 2000203)
Cuatrimestre	Numérico	NO	Cuatrimestre, primero o segundo, en que el alumno cursa la asignatura
Turno	Alfanumérico (10)	NO	Turno, mañana o tarde, en que el alumno cursa la

			asignatura
Num_convocatoria (PK)	Numérico	NO	Numero de convocatoria de la asignatura, en la que se encuentra el alumno en el año académico.
Num_dispensas	Numérico	NO	Número de veces que el alumno ha dispensado justificadamente la asignatura hasta ese año académico
Num_libres_dispensas	Numérico	NO	Número de veces que el alumno ha dispensado injustificadamente la asignatura hasta ese año académico
Nota	Decimal (2,2)	NO	Nota del alumno en la asignatura en el año académico
Libre_eleccion	Alfanumérico (2)	SI	Toma valor s el alumno cursa la asignatura como libre elección

```
CREATE TABLE CURSA(
DNI                    varchar(9)NOT NULL,
ID_ASIGNATURA          int  NOT NULL,
ANIO_ACADEMICO         int  NOT NULL,
CUATRIMESTRE           int  NOT NULL,
TURNO                  varchar(10) NOT NULL,
NUM_CONVOCATORIA       int  NOT NULL,
NUM_DISPENSAS          int NOT NULL,
NUM_LIBRES_DISPENSAS int NOT NULL,
NOTA                   decimal(4,2) NOT NULL,
LIBRE_ELECCION         varchar(2),
PRIMARY KEY (DNI,ID_ASIGNATURA, ANIO_ACADEMICO, NUM_CONVOOCATORIA),
FOREIGN KEY (DNI) REFERENCES ALUMNO,
FOREIGN KEY (ID_ASIGNATURA) REFERENCES ASIGNATURA);
```

Tabla 15. Tabla Cursa

TABLA	DEPARTAMENTO		
NOMBRE DEL CAMPO	**TIPO DE DATOS**	**ADMITE NULOS**	**DESCRIPCIÓN**
Id_departamento (PK)	Numérico	NO	Identificador del departamento al que puede pertenecer una asignatura
Ds_departamento	Alfanumérico (70)	NO	Nombre del departamento
Id_cat_departamento (FK)	Numérico	NO	Identificador de la categoría a la que pertenece el departamento

```
CREATE TABLE DEPARTAMENTO(
ID_DEPARTAMENTO        int PRIMARY KEY,
DS_DEPARTAMENTO        varchar(70) NOT NULL,
ID_CAT_DEPARTAMENTO            int NOT NULL,
FOREIGN KEY (ID_CAT_DEPARTAMENTO) REFERENCES CATEGORIA_DEPARTAMENTO);
```

Tabla 16. Tabla Departamento

TABLA	DOCENTE		
NOMBRE DEL CAMPO	**TIPO DE DATOS**	**ADMITE NULOS**	**DESCRIPCIÓN**
DNI (PK) (FK)	Alfanumérico (9)	NO	DNI del docente
Id_cat_laboral (FK)	Numérico	NO	Categoría profesional del docente en la universidad

```
CREATE TABLE DOCENTE(
DNI                    varchar(9) PRIMARY KEY,
ID_CAT_LABORAL         int NOT NULL,
FOREIGN KEY (DNI) REFERENCES PERSONA
          ON DELETE CASCADE,
FOREIGN KEY (ID_CAT_LABORAL) REFERENCES CATEGORIA_LABORAL);
```

Tabla 17. Tabla Docente

TABLA	ESTADO		
NOMBRE DEL CAMPO	**TIPO DE DATOS**	**ADMITE NULOS**	**DESCRIPCIÓN**
Id_estado (PK)	Numérico	NO	Identificador del estado de un titulación
Ds_estado	Alfanumérico (25)	NO	Descripción del estado. Puede ser "Vigente" cuando la titulación se encuentra en el plan de estudios vigente, "En extinción" si la titulación no pertenece al plan de estudios pero aún se imparte o "Extinguida" cuando deja de impartirse en la universidad

```
CREATE TABLE ESTADO(
ID_ESTADO   int PRIMARY KEY,
DS_ESTADO  varchar(15) NOT NULL);
```

Tabla 18. Tabla Estado

TABLA	FINANCIACION		
NOMBRE DEL CAMPO	**TIPO DE DATOS**	**ADMITE NULOS**	**DESCRIPCIÓN**
Id_financiacion (PK)	Numérico	NO	Identificador de la financiación de una universidad
Ds_financiacion	Alfanumérico (25)	NO	Descripción de la financiación, puede ser "Pública" o "Privada"

```
CREATE TABLE FINANCIACION(
ID_FINANCIACION          int PRIMARY KEY,
DS_FINANCIACION          varchar(25) NOT NULL);
```

Tabla 19. Tabla Financiación

TABLA	GRUPO_ESTUDIO		
NOMBRE DEL CAMPO	**TIPO DE DATOS**	**ADMITE NULOS**	**DESCRIPCIÓN**
Id_grupo_estudio (PK)	Numérico	NO	Identificador del grupo de estudios al que pertenece una titulación
Ds_grupo_estudio	Alfanumérico (50)	NO	Nombre del grupo de estudios que obedece a la rama de la ciencia al que pertenece, humanidades y filosofía, ingeniería, historia…

```
CREATE TABLE GRUPO_ESTUDIO(
ID_GRUPO_ESTUDIO      int PRIMARY KEY,
DS_GRUPO_ESTUDIO      varchar(50) NOT NULL);
```

Tabla 20. Tabla Grupo estudio

TABLA	IMPARTE		
NOMBRE DEL CAMPO	**TIPO DE DATOS**	**ADMITE NULOS**	**DESCRIPCIÓN**
DNI (PK) (FK)	Alfanumérico (9)	NO	DNI del docente que imparte la asignatura en el año académico
Id_asignatura (PK) (FK)	Numérico	NO	Asignatura impartida por el docente en el año académico
Anio_academico (PK)	Numérico	NO	Año académico en el que se imparte la asignatura
Nota_docencia	Decimal (2,4)	NO	Nota que obtiene el profesor y evalúa su docencia en la asignatura y año académico

```
CREATE TABLE IMPARTE(
DNI                   varchar(9) NOT NULL,
ID_ASIGNATURA         int NOT NULL,
ANIO_ACADEMICO        int NOT NULL,
NOTA_DOCENCIA         decimal(4,2) NOT NULL,
PRIMARY KEY (ID_ASIGNATURA,DNI, ANIO_ACADEMICO),
FOREIGN KEY (ID_ASIGNATURA) REFERENCES ASIGNATURA,
FOREIGN KEY (DNI) REFERENCES DOCENTE);
```

Tabla 21. Tabla Imparte

TABLA	MATRICULA		
NOMBRE DEL CAMPO	**TIPO DE DATOS**	**ADMITE NULOS**	**DESCRIPCIÓN**
Id_centro (PK) (FK)	Numérico	NO	Identificador del centro en el que se matricula un alumno
Id_titulacion (PK) (FK)	Numérico	NO	Identificador de la titulación en el que se matricula un alumno
DNI (PK) (FK)	Alfanumérico (9)	NO	DNI del alumno matriculado en la titulación y centro
Anio_academico	Numérico	NO	Año académico de la matriculación en la titulación y centro por el alumno
Convocatoria	Numérico	NO	Convocatoria, ordinaria o extraordinaria, en la que se oferta la titulación en el centro, para la que se corresponden las notas de acceso y en la que accede el alumno
Anio_academico_fin	Numérico	NO	Año académico en el que el alumno concluye la titulación
Plazas	Numérico	NO	Número de plazas ofertadas para la titulación en el año académico, centro y convocatoria
Nota_acceso_PAU	Decimal (2,2)	NO	Nota de acceso para alumnos cuyo tipo de acceso es “PAU”, para la titulación en el año académico, centro y convocatoria
Nota_acceso_FP	Decimal (2,2)	NO	Nota de acceso para alumnos cuyo tipo de acceso es “Formación Profesional”, para la titulación en el año académico, centro y convocatoria

Nota_acceso_25	Decimal (2,2)	NO	Nota de acceso para alumnos cuyo tipo de acceso es “Mayores de 25”, para la titulación en el año académico, centro y convocatoria
Tasa_credito	Decimal (5,2)	NO	Tasa del crédito de las asignaturas de la titulación en el año académico que debe abonar el alumno
Tipo_acceso	Alfanumérico (15)	NO	Tipo de acceso del alumno, prueba general, por formación superior o por el acceso a mayores de 25 años
Nota_acceso	Decimal (2,2)	NO	Nota con la que accede a la titulación en el centro

```
CREATE TABLE MATRICULA(
ID_CENTRO              int NOT NULL,
ID_TITULACION          int NOT NULL,
DNI                    varchar(9) NOT NULL,
ANIO_ACADEMICO         int NOT NULL,
CONVOCATORIA           int NOT NULL,
ANIO_ACADEMICO_FIN     int NOT NULL,
PLAZAS                 int  NOT NULL,
NOTA_ACCESO_PAU        decimal(4,2) NOT NULL,
NOTA_ACCESO_FP         decimal(4,2) NOT NULL,
NOTA_ACCESO_25         decimal(4,2) NOT NULL,
TASA_CREDITO           decimal(5,2) NOT NULL,
TIPO_ACCESO            varchar(15) NOT NULL,
NOTA_ACCESO            decimal(4,2) NOT NULL,
PRIMARY KEY (ID_CENTRO,ID_TITULACION,DNI),
FOREIGN KEY (ID_CENTRO) REFERENCES CENTRO,
FOREIGN KEY (ID_TITULACION) REFERENCES TITULACION,
FOREIGN KEY (DNI) REFERENCES PERSONA);
```

Tabla 22. Tabla Matricula

TABLA	PERSONA		
NOMBRE DEL CAMPO	**TIPO DE DATOS**	**ADMITE NULOS**	**DESCRIPCIÓN**
DNI (PK)	Alfanumérico (9)	NO	DNI que identifica a cada persona
Nombre	Alfanumérico (30)	NO	Nombre de la persona
Apellido_1	Alfanumérico (30)	NO	Apellido principal de la persona
Apellido_2	Alfanumérico (30)	NO	Segundo apellido de la persona
Anio_nacimiento	Numérico	NO	Año de nacimiento de la persona
Sexo	Alfanumérico (5)	NO	Género, hombre o mujer, de la persona
Tipo	Alfanumérico (10)	NO	Especifica si la persona es un alumno o un docente

```
CREATE TABLE PERSONA(
DNI                varchar(9) PRIMARY KEY,
NOMBRE             varchar(30) NOT NULL,
APELLIDO_1         varchar(30) NOT NULL,
APELLIDO_2         varchar(30) NOT NULL,
ANIO_NACIMIENTO    int NOT NULL,
SEXO               varchar(5) NOT NULL,
TIPO               varchar (10) NOT NULL);
```

Tabla 23. Tabla Persona

TABLA	TIPO_ASIGNATURA		
NOMBRE DEL CAMPO	**TIPO DE DATOS**	**ADMITE NULOS**	**DESCRIPCIÓN**
Id_tipo_asignatura (PK)	Numérico	NO	Identificador del tipo de asignatura que una dada puede ser
Ds_tipo_asignatura	Alfanumérico (30)	NO	Nombre del tipo de asignatura

```
CREATE TABLE TIPO_ASIGNATURA(
ID_TIPO_ASIGNATURA    int PRIMARY KEY,
DS_TIPO_ASIGNATURA    varchar(30) NOT NULL);
```

Tabla 24. Tabla Tipo asignatura

TABLA	TIPO_TITULACION		
NOMBRE DEL CAMPO	**TIPO DE DATOS**	**ADMITE NULOS**	**DESCRIPCIÓN**
Id_tipo_titulacion (PK)	Numérico	NO	Identificador del tipo de titulación que una dada puede ser
Ds_tipo_titulacion	Alfanumérico (50)	NO	Descripción del tipo de titulación, grado, ingeniería, ingeniería técnica, licenciatura…
CREATE TABLE TIPO_TITULACION(**ID_TIPO_TITULACION int PRIMARY KEY,** **DS_TIPO_TITULACION varchar(50) NOT NULL);**			

Tabla 25. Tabla Tipo titulación

TABLA	TITULACION		
NOMBRE DEL CAMPO	**TIPO DE DATOS**	**ADMITE NULOS**	**DESCRIPCIÓN**
Id_titulacion (PK)	Numérico	NO	Identificador de la titulación
Ds_titulacion	Alfanumérico (100)	NO	Nombre de la titulación
Id_tipo_titulacion (PK) (FK)	Numérico	NO	Identificador del tipo de la titulación
Id_grupo_estudio (PK) (FK)	Numérico	NO	Identificador del grupo de estudio de la titulación
Id_estado (PK) (FK)	Numérico	NO	Identificador del estado de la titulación
Créditos_titulacion	Numérico	NO	Número total de créditos ECTS de la titulación
Num_max_conv	Numérico	NO	Número máximo de convocatorias que tiene un alumno para aprobar cualquier asignatura de la titulación
Fecha_imp	Numérico	NO	Fecha de implantación de la titulación

Fecha_fin	Numérico	NO	Último año académico en el que se imparte la titulación y cualquiera de sus asignaturas. Si toma valor 999912 significa que la titulación aún se imparte y su estado es "Vigente" o "En extinción"

```
CREATE TABLE TITULACION(
ID_TITULACION          int PRIMARY KEY,
DS_TITULACION          varchar(100) NOT NULL,
ID_TIPO_TITULACION     int NOT NULL,
ID_GRUPO_ESTUDIO       int NOT NULL,
CREDITOS_TITULACION    int NOT NULL,
NUM_MAX_CONV           int NOT NULL,
FECHA_IMP              int NOT NULL,
FECHA_FIN              int NOT NULL,
ID_ESTADO              int NOT NULL,
FOREIGN KEY (ID_TIPO_TITULACION) REFERENCES TIPO_TITULACION,
FOREIGN KEY (ID_GRUPO_ESTUDIO)  REFERENCES GRUPO_ESTUDIO,
FOREIGN KEY (ID_ESTADO)  REFERENCES ESTADO);
```

Tabla 26. Tabla Titulación

TABLA	UNIVERSIDAD		
NOMBRE DEL CAMPO	**TIPO DE DATOS**	**ADMITE NULOS**	**DESCRIPCIÓN**
Id_universidad (PK)	Numérico	NO	Identificador de la universidad
Ds_universidad	Alfanumérico (100)	NO	Nombre de la universidad
Id_financiacion (PK) (FK)	Numérico	NO	Identificador del tipo de financiación que recibe la universidad

```
CREATE TABLE UNIVERSIDAD(
ID_UNIVERSIDAD          int PRIMARY KEY,
DS_UNIVERSIDAD          varchar(100) NOT NULL,
ID_FINANCIACION         int NOT NULL,
FOREIGN KEY (ID_FINANCIACION) REFERENCES FINANCIACION);
```

Tabla 27. Tabla Universidad

5.1.2. Inserción de datos de prueba

La inserción de los datos en ODUNI se lleva a cabo a través de una *ETL* realizada con Microsoft Excel.

Los datos incluidos en el sistema no pretenden reflejar una realidad ni información verídica. Su finalidad es comprobar el resultado de la *ETL* que convierte esquemas *E/R* en *esquemas en estrella*. Son, en su mayoría, ficticios. Algunos de carácter público como las asignaturas de las titulaciones de grado ofertadas por la Universidad Carlos III de Madrid o los departamentos de ésta, han sido extraídos de la web.

La fase de extracción de la información de esta *ETL* utiliza como origen de datos las web y los datos aleatorios generados con funciones de Excel. La fase de transformación se implementa a través de funciones de la aplicación, filtros y macros en determinados casos. La carga se produce en scripts de extensión .sql para su posterior ejecución.

El proceso de inserción de datos comienza con la recopilación de información de enlaces web: nombre de universidades, campus, centros, titulaciones, asignaturas y departamentos.

Posteriormente, se crean tablas en un fichero Excel, análogas a las implementadas en ODUNI, y se completan con la información recopilada. Los campos de información sensible, como el DNI de las personas, el año de nacimiento o las notas de evaluaciones son números semi-aleatorios, es decir, los números se generan dentro de un rango conocido y existe un control sobre ellos. Así, por ejemplo, una persona que accede a la Universidad por la vía de mayores de 25 años tiene esa edad o más el año académico que se matricula. La nota estará comprendida entre 0 y 10. Además en el caso de las notas se fuerza a que el porcentaje de aprobadas sea mayor al de notas suspensas.

Las tablas que contienen un histórico de información como Matricula, Cursa o Imparte y que obedecen a una combinación de campos de otras tablas (Asignatura, Alumno, Docente…) están completadas con datos igualmente semi-aleatorios pero añadiendo más controles (usando maros en ocasiones) que simulan check´s y triger, e impiden incoherencias entre los registros de diferentes tablas, como que un alumno matriculado de una titulación curse asignaturas de otra, un profesor imparta asignaturas sin alumnos, no se cumplan las claves primarias y ajenas, etc.

Las claves primarias de todas las tablas, que no son además claves ajenas, son autonuméricos.

Un vez están completas las tablas, se concatenan los campos de cada una, junto con literales propios de la sintaxis de SQL Server y se realizan pequeñas modificaciones en los campos de tipo numérico para sustituir las comas por puntos. El resultado es un script de la inserción de una tupla. Todos los script pueden trasladarse a un script general de inserciones ejecutable, que cargue ODUNI y complete esta primera *ETL*.

5.1.3. Creación del esquema en estrella

Las tablas de este esquema quedan definidas en el capítulo 4. Su implementación se lleva a cabo a través de sentencias de creación de vistas en SQL Server.

Se pueden definir dos fases principales en la implementación del esquema, la primera implementa las *tablas de dimensiones* y la segunda las *tablas de hechos*.

Las *tablas de dimensiones* que presentaban jerarquías y que por tanto enlazaban diferentes tablas del esquema *E/R* se crean utilizando JOINS. Estas tablas, junto con las de dimensiones que se originan de una única tabla que no surge de una relación en el esquema *E/R*, están normalizadas por lo que no es preciso hacer un tratamiento para eliminar duplicados y las *claves subrogadas* (SK) de la dimensión son las mismas que las claves primarias de las tablas. Las *claves subrogas* en un *esquema en estrella* suelen ser autonuméricos que no tienen por qué coincidir con la clave del mismo elemento en el esquema *E/R* inicial.

Para las *dimensiones* que se originaban a partir de tablas surgidas de relaciones del esquema *E/R*, (como el tiempo, que se extraía de Matrícula, Cursa o Imparte) no basta con realizar una consulta que origina la vista, los datos deben agruparse para que la *dimensión* solo tenga una fila por elemento y en algunos casos incluir una transformación para generar la clave subrogada (en la dimensión tiempo la SK es la concatenación de año académico y cuatrimestre)

SCRIPS Tablas de dimensiones esquema en estrella ODUNI

```
-- Dimensión TIEMPO
CREATE VIEW DIM_TIEMPO AS
	(SELECT convert(varchar (6),ANIO_ACADEMICO)+ convert(varchar (1),CUATRIMESTRE) AS
SK_TIEMPO, ANIO_ACADEMICO, CUATRIMESTRE
	FROM CURSA
	GROUP BY ANIO_ACADEMICO, CUATRIMESTRE)

-- Dimensión UNIVERSIDAD
CREATE VIEW DIM_UNIVERSIDAD AS
	(SELECT F.ID_FINANCIACION, DS_FINANCIACION, U.ID_UNIVERSIDAD, DS_UNIVERSIDAD,
CA.ID_CAMPUS, DS_CAMPUS, CE.ID_CENTRO AS SK_CENTRO, DS_CENTRO
	FROM FINANCIACION F,UNIVERSIDAD U,CAMPUS CA,CENTRO CE
	WHERE F.ID_FINANCIACION =U.ID_FINANCIACION AND
U.ID_UNIVERSIDAD=CA.ID_UNIVERSIDAD AND CA.ID_CAMPUS=CE.ID_CAMPUS)

-- Dimensión TITULACION
CREATE VIEW DIM_TITULACION AS
	(SELECT T.ID_TIPO_TITULACION,DS_TIPO_TITULACION,T.ID_GRUPO_ESTUDIO,
DS_GRUPO_ESTUDIO, T.ID_TITULACION, DS_TITULACION,
```

```
                    ASI.ID_ASIGNATURA AS SK_ASIGNATURA, ASI.DS_ASIGNATURA
          FROM TIPO_TITULACION TT,GRUPO_ESTUDIO GE,TITULACION T, ASIGNATURA ASI
          WHERE TT.ID_TIPO_TITULACION =T.ID_TIPO_TITULACION AND
GE.ID_GRUPO_ESTUDIO=T.ID_GRUPO_ESTUDIO AND T.ID_TITULACION=ASI.ID_TITULACION)

-- Dimensión TIPO_ASIGNATURA
CREATE VIEW DIM_TIPO_ASIGNATURA AS
          (SELECT ID_TIPO_ASIGNATURA AS SK_TIPO_ASIGNATURA, DS_TIPO_ASIGNATURA
          FROM TIPO_ASIGNATURA)

-- Dimensión CURSO
CREATE VIEW DIM_CURSO AS
          (SELECT CURSO AS SK_CURSO
          FROM ASIGNATURA
          GROUP BY CURSO)

-- Dimensión DEPARTAMENTO
CREATE VIEW DIM_DEPARTAMENTO AS
          (SELECT CD.ID_CAT_DEPARTAMENTO,DS_CAT_DEPARTAMENTO,D.ID_DEPARTAMENTO AS
SK_DEPARTAMENTO, D.DS_DEPARTAMENTO
          FROM DEPARTAMENTO D, CATEGORIA_DEPARTAMENTO CD
          WHERE CD.ID_CAT_DEPARTAMENTO=D.ID_CAT_DEPARTAMENTO)

-- Dimensión TURNO
CREATE VIEW DIM_TURNO AS
          (SELECT TURNO AS SK_TURNO
          FROM CURSA
          GROUP BY TURNO)

-- Dimensión TIPO_ACCESO
CREATE VIEW DIM_TIPO_ACCESO AS
          (SELECT TIPO_ACCESO AS SK_TIPO_ACCESO
          FROM MATRICULA
          GROUP BY TIPO_ACCESO)

-- Dimensión CONVOCATORIA_ACCESO
CREATE VIEW DIM_CONVOCATORIA_ACCESO AS
          (SELECT CONVOCATORIA AS SK_CONVOCATORIA_ACCESO
          FROM MATRICULA
          GROUP BY CONVOCATORIA)

-- Dimensión CONVOCATORIA_EXAMEN
CREATE VIEW DIM_CONVOCATORIA AS
          (SELECT NUM_CONVOCATORIA AS SK_CONVOCATORIA
          FROM CURSA
          GROUP BY NUM_CONVOCATORIA)

-- Dimensión CAT_LABORAL
CREATE VIEW DIM_CAT_LABORAL AS
          (SELECT ID_CAT_LABORAL AS SK_CAT_LABORAL, DS_CAT_LABORAL
```

```
        FROM CATEGORIA_LABORAL)

-- Dimensión SEXO_ALUMNO
CREATE VIEW DIM_SEXO_ALUMNO AS
        (SELECT SEXO AS SK_SEXO_ALUMNO
        FROM PERSONA
        WHERE TIPO='ALUMNO'
        GROUP BY SEXO)

-- Dimensión SEXO_DOCENTE
CREATE VIEW DIM_SEXO_DOCENTE AS
        (SELECT SEXO AS SK_SEXO_DOCENTE
        FROM PERSONA
        WHERE TIPO='DOCENTE'
        GROUP BY SEXO)

-- Dimensión ESTADO
CREATE VIEW DIM_ESTADO AS
        (SELECT ID_ESTADO AS SK_ESTADO, DS_ESTADO
        FROM ESTADO)

-- Maestro de Alumnos
CREATE VIEW MST_ALUMNO AS
        (SELECT P.DNI AS SK_DOCENTE, P.NOMBRE, P.APELLIDO_1, P.APELLIDO_2
        FROM PERSONA P
        WHERE TIPO='ALUMNO')

-- Maestro de Docentes
CREATE VIEW MST_DOCENTE AS
        (SELECT P.DNI AS SK_DOCENTE, P.NOMBRE, P.APELLIDO_1, P.APELLIDO_2
        FROM PERSONA P
        WHERE TIPO='DOCENTE')
```

Tabla 28. SCRIPS Tablas de dimensiones esquema en estrella ODUNI

La fase de creación de las *tablas de hechos* incluye una serie de vistas que se generan secuencialmente, alimentándose unas de otras hasta llegar a generar HECHOS_ALUMNO y HECHOS_DOCENTE.

En estas vistas se extraen los identificadores de las dimensiones, las medidas de hechos y los identificadores necesarios para el proceso, y así llegar a conformar las *tablas de hechos*. Las agrupaciones aseguran la unicidad de la clave de las vistas finales.

El nombre de las vistas hace referencia al orden de la ejecución de las transformaciones y la información principal que aportan.

SCRIPS Tablas de hechos esquema en estrella ODUNI

```
--- Primera aproximación a HECHOS_ALUMNO
-- En esta vista la información se relativa a las dimensiones más relacionadas con la asignatura y las medidas de la relacion CURSA se agrupan en una tabla
CREATE VIEW TRS1_CURSA_ASIGNATURA AS
(SELECT
    convert(varchar (6),C.ANIO_ACADEMICO)+ convert(varchar (1),C.CUATRIMESTRE) AS SK_TIEMPO,
    C.ANIO_ACADEMICO, --Campo necesario para uniones posteriores
    C.CUATRIMESTRE, --Campo necesario para uniones posteriores
    C.ID_ASIGNATURA AS SK_ASIGNATURA,
    C.TURNO AS SK_TURNO,
    C.NUM_CONVOCATORIA AS SK_CONVOCATORIA,
    C.DNI AS SK_ALUMNO,
    A.ID_TITULACION, --Campo necesario para en uniones posteriores acceder a la info.de la titulación
    A.CURSO AS SK_CURSO,
    A.ID_TIPO_ASIGNATURA AS SK_TIPO_ASIGNATURA,
    A.ID_DEPARTAMENTO AS SK_DEPARTAMENTO,
    -- Medidas aportadas por CURSA
    --En realidad no se producen agregaciones de datos a este nivel, pero por el empleo del GROUP BY (asegura la no duplicidad de claves) por él se precisan los operadores (pueden emplearse sum, min o max)
    MIN(A.CREDITOS) AS CREDITOS,
    SUM(C.NOTA) AS IN_NOTA,
    MAX(C.NUM_DISPENSAS) AS IN_DISPENSAS,
    MAX(C.NUM_LIBRES_DISPENSAS) AS IN_LIBRES_DISPENSAS
FROM CURSA C, ASIGNATURA A
WHERE C.ID_ASIGNATURA=A.ID_ASIGNATURA
GROUP BY C.ANIO_ACADEMICO, C.CUATRIMESTRE, C.ID_ASIGNATURA, C.TURNO, C.NUM_CONVOCATORIA, C.DNI, A.ID_TITULACION, A.CURSO, A.ID_DEPARTAMENTO, A.ID_TIPO_ASIGNATURA)


--- Segunda aproximación a HECHOS_ALUMNO
-- Se añade el estado de la titulación
CREATE VIEW TRS2_TITULACION AS
(SELECT TRS1.*, T.ID_ESTADO AS SK_ESTADO
FROM TRS1_CURSA_ASIGNATURA TRS1, TITULACION T
WHERE TRS1.ID_TITULACION=T.ID_TITULACION)


--- Tercera aproximación a HECHOS_ALUMNO
-- Se añade la información referente a la matricula del alumno, y las medidas que aporta est relación MATRICULA
CREATE VIEW TRS3_MATRICULA AS
(SELECT TRS2.*, M.ID_CENTRO AS SK_CENTRO,
        M.CONVOCATORIA AS SK_CONVOCATORIA_ACCESO,
        M.TIPO_ACCESO AS SK_TIPO_ACCESO,
        -- Medidas aportadas por MATRICULA
        (M.TASA_CREDITO*TRS2.CREDITOS) AS IN_COSTE_TASAS,--se precalcula el coste de la
```

```
asignatura
        M.NOTA_ACCESO AS IN_NOTA_ACCESO
FROM TRS2_TITULACION TRS2, MATRICULA M
WHERE  TRS2.SK_ALUMNO=M.DNI AND
        TRS2.ID_TITULACION=M.ID_TITULACION)

SELECT * FROM TRS3_MATRICULA

--- Cuarta aproximación a HECHOS_ALUMNO
-- Se añade la información referente al alumno, y la medida edad del alumno
CREATE VIEW TRS4_ALUMNO AS
(SELECT TRS3.*,
        P.SEXO AS SK_SEXO_ALUMNO,
        ((YEAR(GETDATE())) - P.ANIO_NACIMIENTO) AS IN_EDAD_ALUMNO
FROM TRS3_MATRICULA TRS3, PERSONA P
WHERE TRS3.SK_ALUMNO=P.DNI)


--- Quinta aproximación a HECHOS_ALUMNO
-- Esta vista añade el DNI del docente para usarlo en cruces de datos posteriores
-- Duplica, aquellos registros, tantas veces como profesores diferentes haya tenido una asignatura en
un año y cuatrimestre
CREATE VIEW TRS5_DOCENTE AS
(SELECT TRS4.*,
        I.DNI AS DNI
FROM TRS4_ALUMNO TRS4, IMPARTE I
WHERE  TRS4.SK_ASIGNATURA=I.ID_ASIGNATURA AND
        TRS4.ANIO_ACADEMICO=I.ANIO_ACADEMICO)


--- Sexta aproximación a HECHOS_ALUMNO
-- Se añade la información de las dimensiones más relacionadas con el docente, el sexo y la categoría
laboral
CREATE VIEW TRS6_INFO_DOCENTE AS
(SELECT TRS5.*,
        D.ID_CAT_LABORAL AS SK_CAT_LABORAL,
        P.SEXO AS SK_SEXO_DOCENTE
FROM TRS5_DOCENTE TRS5, DOCENTE D, PERSONA P
WHERE  TRS5.DNI=D.DNI AND
        D.DNI=P.DNI)


--- Creación de HECHOS_ALUMNO
-- Se realizan las agrupaciones para eliminar los registros duplicados por las asignatura con varios
profesores (por año académico)
-- El nivel de profesor (SK_DOCENTE) se pierde, llegará por la tabla de HECHOS_DOCENTE, al no
agrupar por DNI
CREATE VIEW HECHOS_ALUMNO AS
(SELECT --Identificadores, unión de todas las claves subrogadas --> PK de HECHOS_ALUMNOS
        SK_ALUMNO,
```

```
        SK_TIEMPO,
        SK_CENTRO,
        SK_ASIGNATURA,
        SK_TIPO_ASIGNATURA,
        SK_CURSO,
        SK_DEPARTAMENTO,
        SK_TURNO,
        SK_CONVOCATORIA,
        SK_TIPO_ACCESO,
        SK_CONVOCATORIA_ACCESO,
        SK_CAT_LABORAL,
        SK_SEXO_ALUMNO,
        SK_SEXO_DOCENTE,
        SK_ESTADO,
        -- Medidas
        MAX(IN_NOTA) AS IN_NOTA,
        MAX(IN_DISPENSAS) AS IN_DISPENSAS,
        MAX(IN_LIBRES_DISPENSAS) AS IN_LIBRES_DISPENSAS,
        MAX(IN_COSTE_TASAS) AS IN_COSTE_TASAS,
        MAX(IN_NOTA_ACCESO) AS IN_NOTA_ACCESO,
        MAX(IN_EDAD_ALUMNO) AS IN_EDAD_ALUMNO
FROM TRS6_INFO_DOCENTE
GROUP BY SK_ALUMNO, SK_TIEMPO, SK_CENTRO, SK_ASIGNATURA,
SK_TIPO_ASIGNATURA,SK_CURSO, SK_DEPARTAMENTO, SK_TURNO, SK_CONVOCATORIA,
SK_TIPO_ACCESO,
        SK_CONVOCATORIA_ACCESO, SK_CAT_LABORAL, SK_SEXO_ALUMNO, SK_SEXO_DOCENTE,
SK_ESTADO)

-------------------------------------------------------------------------------
-------------------------------------------------------------------------------

--- Primera aproximación a HECHOS_DOCENTE
-- Partiendo de la tabla TRS4_ALUMNO se obtendrá la nota del docente y gregará la información en
función de las necesidades de HECHOS_DOCENTE, perdiendo el nivel de alumno (SK_ALUMNO)
CREATE VIEW TRS1_AGG_ALUMNOS AS
(SELECT TRS4.*,
        I.DNI AS SK_DOCENTE,
        I.NOTA_DOCENCIA AS IN_NOTA_DOCENTE
FROM TRS4_ALUMNO TRS4, IMPARTE I
WHERE  TRS4.SK_ASIGNATURA=I.ID_ASIGNATURA AND
        TRS4.ANIO_ACADEMICO=I.ANIO_ACADEMICO)


--- Segunda aproximación a HECHOS_ALUMNO
-- Se añade la información de las dimensiones más relacionadas con el docente, el sexo y la categoría
laboral (es la misma acción que creó la vista TRS6_INFO_DOCENTE)
CREATE VIEW TRS2_INFO_DOCENTE AS
(SELECT TRS1.*,
        D.ID_CAT_LABORAL AS SK_CAT_LABORAL,
        P.SEXO AS SK_SEXO_DOCENTE,
```

```
        ((YEAR(GETDATE())) - P.ANIO_NACIMIENTO) AS IN_EDAD_DOCENTE
FROM TRS1_AGG_ALUMNOS TRS1, DOCENTE D, PERSONA P
WHERE  TRS1.SK_DOCENTE=D.DNI AND
        D.DNI=P.DNI)


--- Creación de HECHOS_DOCENTE
CREATE VIEW HECHOS_DOCENTE AS
(SELECT --Identificadores, unión de todas las claves subrogadas --> PK de HECHOS_DOCENTE
        SK_DOCENTE,
        SK_TIEMPO,
        SK_CENTRO,
        SK_ASIGNATURA,
        SK_TIPO_ASIGNATURA,
        SK_CURSO,
        SK_DEPARTAMENTO,
        SK_TURNO,
        SK_CONVOCATORIA,
        SK_TIPO_ACCESO,
        SK_CONVOCATORIA_ACCESO,
        SK_CAT_LABORAL,
        SK_SEXO_ALUMNO,
        SK_SEXO_DOCENTE,
        SK_ESTADO,
        -- Medidas
        MAX(IN_NOTA_DOCENTE) AS IN_NOTA_DOCENTE,
        MAX(IN_EDAD_DOCENTE) AS IN_EDAD_DOCENTE
FROM TRS2_INFO_DOCENTE
GROUP BY SK_DOCENTE, SK_TIEMPO, SK_CENTRO, SK_ASIGNATURA,
SK_TIPO_ASIGNATURA,SK_CURSO, SK_DEPARTAMENTO, SK_TURNO, SK_CONVOCATORIA,
SK_TIPO_ACCESO,
        SK_CONVOCATORIA_ACCESO, SK_CAT_LABORAL, SK_SEXO_ALUMNO, SK_SEXO_DOCENTE,
SK_ESTADO)
```

Tabla 29. SCRIPS Tablas de hechos esquema en estrella ODUNI

5.2. UNICUBE

El *cubo*, definido con el componente de Cognos Transformer, no obtiene la información directamente del modelo *ROLAP* de ODUNI, sino que la extrae de un modelo de datos generado con el componente de Cognos Framework Manager y publicado en Cognos Connection.

Las tablas que son declaradas en el espacio Capa de BBDD son las 16 *tablas de dimensiones* y las dos *tablas de hechos* establecidas en el *esquema en estrella*.

Las tablas definidas en el espacio Capa de Transformer son estas mismas pero algunas de ellas presentan cambios en algunos campos descriptivos. Por ejemplo, la dimensión tiempo, almacena el año académico como el concatenado del primer año y las dos últimas cifras del segundo. Mostrar en el *cubo* 200340 hace poco comprensible la lectura al usuario. En la capa de Transformer, haciendo uso de los operadores definidos en Framework Manager, se separa este campo y se introduce una barra "/", de modo que al *cubo* llegue 2003/04, literal que si entiende el usuario. Realizar estas operaciones en el *framework* libera al motor de base de datos de operaciones que tienen un fin de formato, ya que la información es la misma.

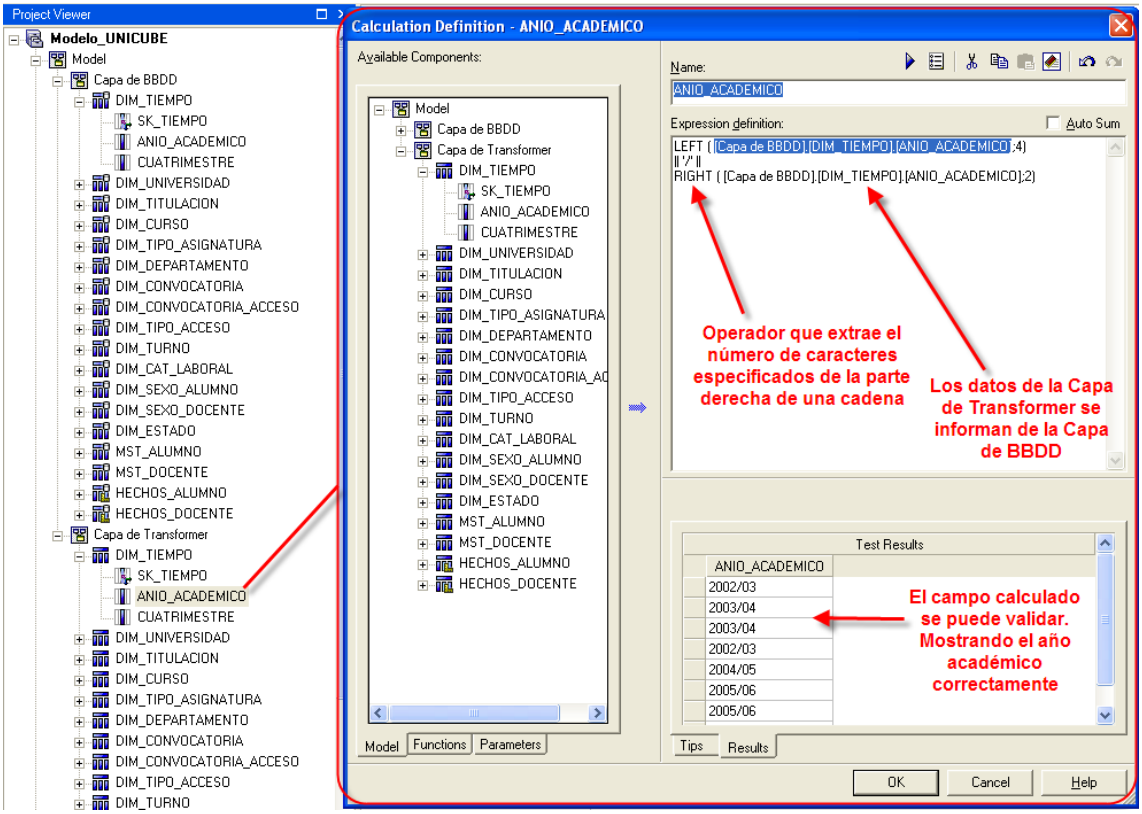


Ilustración 46. Definición consulta en Capa de Transformer

En el *framework* se define el tipo de datos que contiene un campo, pudiendo ser, principalmente:

Identificadores: Son campos que identifican elementos, las *claves subrogadas* del esquema en estrella son de este tipo.

Atributos: campos descriptivos, se utilizarán como etiquetas en el cubo.

Medidas: son los atributos de hechos cuantitativos.

Una vez definidos todos los campos se publica el *framework* como Modelo UNICUBE, en la definición de este paquete de datos solo se permitirá la visibilidad del espacio Capa de Transformer (y de esta capa todas las tablas):

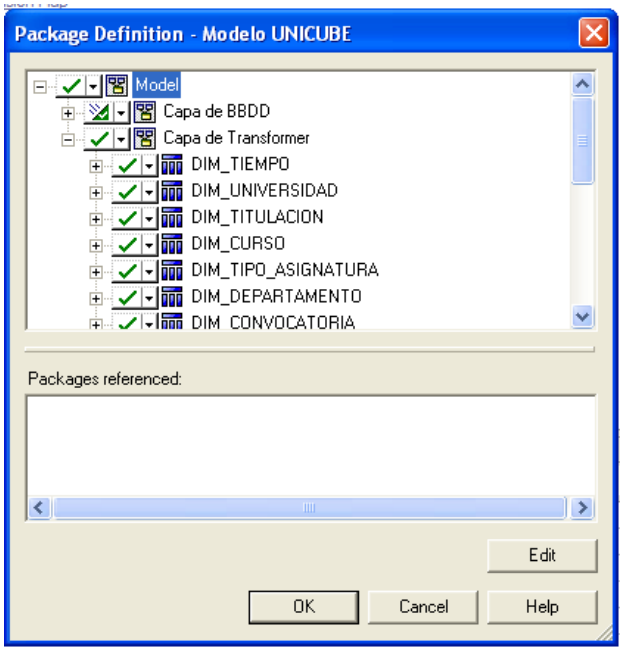


Ilustración 47. Definición paquete Modelo UNICUBE

El *cubo* toma como origen de datos las tablas de Modelo UNICUBE.

En el *modelo del cubo* UNICUBE se deben especificar las propiedades de cada *dimensión*, así como las de sus niveles. Para cada una se especifica:

- Dato (*Source*): es el identificador del elemento. Este es lo mínimo que se debe especificar de una dimensión.

- Etiqueta (*Label*): es la descripción que mostrará el dato. (source: 200304; label: 2003/04)

- Ordenación (*Order by*): ordenación del elemento en el nivel de la *dimensión*. Puede ser orden alfabético o numérico, ascendente o descendente.

- Código de categoría (*Category Code*): Se trata de un código único para cada elemento de un *cubo*, equivale a la clave primaria de una tabla de BD, pero a

nivel global de *cubo*. Es un campo usado por los procesos internos de la aplicación. Este código es generado automáticamente por la herramienta pero puede especificarse, por ejemplo como la concatenación del número de dimensión, número de nivel y clave del nivel (esta fórmula asegura la unicidad de la clave). Puede ser útil definirlo ya que, con frecuencia, las descripciones de los errores que se producen hacen referencia al *category code* del elemento que produce el fallo y si es un número descriptivo es más fácil localizar el error (un *category code* generado automáticamente por el programa no se sabe a qué elemento pertenece)

- Unicidad del campo: La marca de *Unique* establece si se sólo habrá un elemento diferente en el nivel, es decir si se trata del nivel que cruzará con la tabla de hechos.

- Refresco de atributos: Se puede marcar la opción *Refres label* de refrescar los atributos de un elemento como la etiqueta o la descripción. Una vez se lee el dato se queda almacenado en el *cubo*, y en posteriores generaciones de éste la etiqueta o la descripción no se leen de nuevo (y actualizan) si no se marca la opción, por cuestiones de eficiencia. Esta es una opción importante si se produce un número alto de transacciones de modificación en la BD.

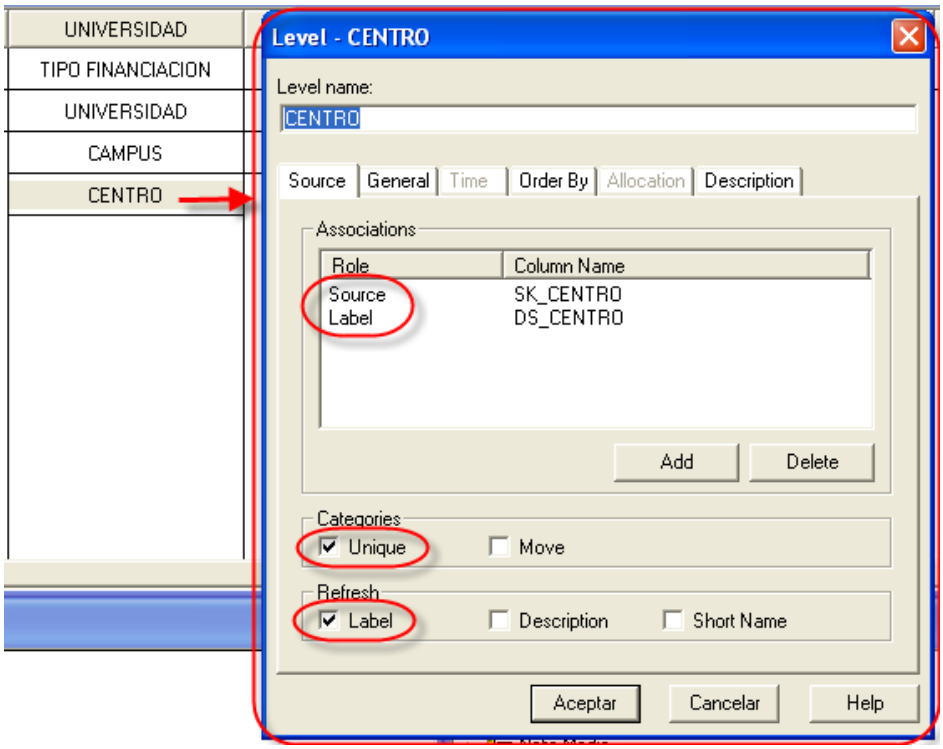


Ilustración 48. Definición 1 nivel de dimensión de UNICUBE

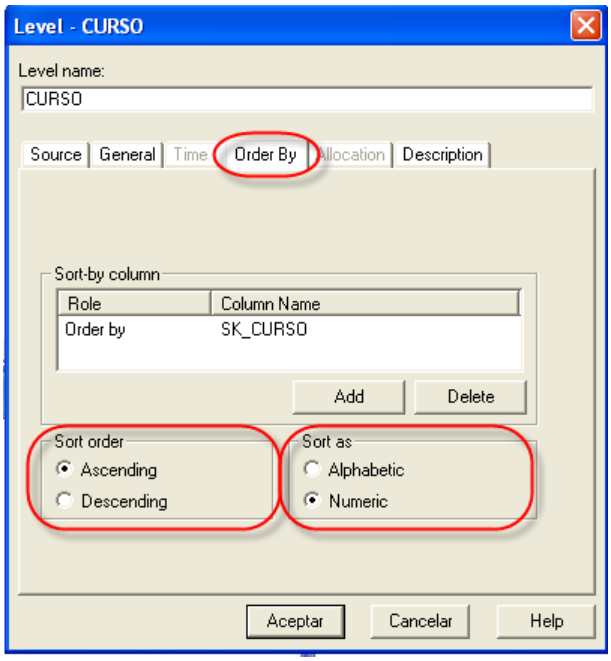


Ilustración 49. Definición 2 nivel de dimensión de UNICUBE

Capítulo 6

En este capítulo tendrá lugar la documentación de las pruebas que verifiquen el correcto funcionamiento del proceso de conversión entre el esquema *E/R* y el *esquema en estrella*. Así mismo, se testeará el correcto funcionamiento del cubo diseñado.

Las comprobaciones deberán asegurar que la información contenida en el modelo inicial no se pierde ni se deteriora en el proceso de conversión, y puede ser analizada en el *cubo*.

6. EXPERIMENTACIÓN Y RESULTADOS

Este capítulo pretende validar el sistema construido, comprobando que los datos del *esquema en estrella*, mostrados en el *cubo* que es alimentado directamente por este esquema, son los mismos que los contenidos en el esquema *E/R*.

Las pruebas deben englobar las tablas de dimensiones (originan las dimensiones del cubo) y las tablas de hechos (contienen las medidas del cubo y las claves de las dimensiones).

Estas pruebas se realizarán mediante comparaciones de dos tipos:

- *Tablas de dimensiones*: comparación entre consultas realizadas con el motor de bases de datos, en el esquema *E/R* de ODUNI y en el *esquema en estrella*.

- *Tablas de hechos*: comparación entre consultas realizadas con el motor de bases de datos en el esquema *E/R* de ODUNI y los análisis equivalentes a dichas consultas en el *cubo* UNICUBE.

  Esta comparación a pesar de no realizarse directamente sobre el *esquema en estrella* en la BD, es válido, ya que el *cubo* toma la información directamente de este esquema presentándola de forma más amigable.

  Pero la elección de esta forma de comparar obedece a una mejora del rendimiento y no de la presentación de los datos.

  La validación de las tablas de hechos consiste en la comparación de las medidas en función de las columnas de la tabla que forman parte de la clave primaria de dicha tabla, es decir, comparar las medidas mediante cruces de dimensiones. Para el esquema *E/R* estas consultas resultan complicadas y pesadas para el motor de BD pero son inevitables para las pruebas. Sin embargo, estas mismas consultas para el *esquema en estrella* pueden realizarse en el *cubo*.

  Los *indicadores* están pre-calculados en ambos sistemas (*tabla de hechos* de la BD y *cubo*) pero el análisis del *cubo* resulta mucho más eficiente porque los orígenes de datos están indizados y el resultado es más rápido. Además los cruces de *dimensiones* en el *cubo* son mucho más sencillos de realizar, ya que la interfaz de la herramienta es mucho más intuitiva y cercana al usuario que las querys en SQL, mejorando también el rendimiento del encargado de pruebas.

  El análisis resultante del *cubo* podría exportarse a una tabla para compararlo con la tabla resultado de la consulta sobre el esquema *E/R*, pero con esta operación se perdería la eficiencia ganada por el hecho de utilizar el *cubo*. Tratar de

comparar absolutamente todas las medidas bajo todas las combinaciones de dimensiones sería un trabajo infinito, por lo que se hará una comparación de una muestra de datos y cruces representativos.

Es interesante anotar que la comparación de las dimensiones no podría hacerse directamente con un análisis de UNICUBE, en contra de lo que ocurre con los hechos. En todo caso podría realizarse una comparación con el *mapa de categorías* del *modelo del cubo* construido.

El *mapa de categorías* muestra todos los elementos de las *dimensiones* existentes en las *tablas de dimensiones* del *esquema en estrella*. Sin embargo estos elementos deben tener medidas con valor para mostrarse en los análisis. Es decir, puede existir la titulación A (elemento de la dimensión titulación) pero si no hay ningún alumno que la curse ni profesor que la imparta no existirán medidas asociadas a este elemento de la dimensión, con lo que no figurará en los análisis del cubo. Si bien, aparecerá en el mapa de categorías.

Para realizar una comparación sobre el *mapa de categorías* esta debería ser visual (no se puede exportar a ningún formato de tipo hoja de cálculo, sobre el que operar) con la posibilidad de cometer un mayor número de errores en la validación.

6.1. Tablas de dimensiones

Existen 16 tablas de dimensiones cuyo contenido debe compararse con el existente en esquema *E/R*. Dado que cada tabla de dimensión del esquema en estrella se crea a partir de una o varias tablas del modelo E/R o de algunos campos de ciertas tablas, cada comprobación tiene asociada una consulta diferente sobre el modelo E/R. En todos los casos, las consultas sobre la tabla de dimensión, del esquema en estrella, será una selección de todos los valores y todos los campos de las tablas.

El conjunto de datos que no forman parte de la intersección, entre el resultado la consulta sobre el modelo E/R y la consulta de la tabla de dimensión correspondiente, debe ser nulo para que la conversión del modelo sea correcta.

Sin embargo, por la definición de la creación de las dimensiones, los valores siempre serán los mismos, ya que se trata de uniones a partir de los identificadores, las consultas que comprueban que una dimensión tiene los valores correctos es la misma que crea la dimensión:

IMPLEMENTACIÓN	COMPROBACIÓN
CREATE VIEW MST_ALUMNO AS (SELECT distinct (P.DNI) AS SK_DOCENTE, P.NOMBRE, P.APELLIDO_1, P.APELLIDO_2	**SELECT** distinct (sk_alumno) FROM **MST_ALUMNO** WHERE sk_alumno NOT IN

FROM PERSONA P WHERE TIPO='ALUMNO')	(SELECT distinct (P.DNI) FROM PERSONA P WHERE TIPO='ALUMNO')

El resto de las tablas de dimensiones se encuentran en la misma situación, por lo que la propia definición del esquema verifica su validez.

6.2. Tablas de hechos

Las columnas de *tablas de hechos* surgen como el conjunto de claves primarias de las *tablas de dimensiones* con quién se relacionan y como columnas adicionales las *medidas* calculadas. Cada fila de la tabla es el producto cartesiano de todas las *dimensiones* relacionadas y las respectivas *medidas* calculadas para los valores que toman las *dimensiones* de una combinación concreta.

Comprobar todos los datos del *esquema en estrella* con las posibles combinaciones, hay 16 *dimensiones*, algunas de ellas jerarquizadas en más de un nivel, es inviable. Sería como comprobar cada tupla de la *tabla de hechos* (hay miles).

En este tipo de tablas todos son valores de riesgo, verificar el valor límite permitido comprueba que el campo es capaz de almacenarlo y el proceso capaz de calcularlo pero no implica que el valor sea correcto. Por ejemplo, una nota académica oscila entre 0 y 10, verificar que se almacena y calcula el 0 y el 10 no implica que sean datos correctos, implica que se calculan y almacenan esos valores.

En el *esquema en estrella*, y por extensión en el *cubo*, las *medidas* se calculan en función del valor de las *dimensiones*. Por tanto, si una *medida* está bien calculada para un elemento de una *dimensión* y el total de esa *dimensión* es igualmente correcto, se puede inferir que la medida analizada está bien calculada para todo elemento de esa *dimensión*. En este sentido se realizan dos tipos de comprobaciones:

- **Total del cubo**: comprobado que el total de cada medida es el mismo en el *cubo* que en el esquema *E/R* se verifica que no se ha producido pérdida de información.

- **Reparto correcto**: si no hay perdida de información se debe verificar que está correctamente "repartida" en las *dimensiones*. Es decir, analizando las *medidas* para un elemento de una *dimensión* del *cubo* se analizan para todos los elementos de la misma *dimensión*.

Para hacer las comprobaciones principales que aseguren la veracidad de los datos se comprobarán las *medidas* de la siguiente forma:

- Analizadas para un elemento de una *dimensión*, si ésta está jerarquizada se desglosa hasta su nivel inferior.

- Analizada para el total de la *dimensión*.

La Tabla 30. Validación datosmuestra las medidas analizadas desde cada dimensión, si el total y un elemento especificado, elegido aleatoriamente (en las dimensiones jerarquizadas se detalla el elemento y su nivel superior, con el fin de identificarlo) concuerdan (se muestra C) la prueba es satisfactoria, si no concuerdan se muestra NC y las pruebas detectan errores en la conversión de esquemas.

Las medidas que no se encuentran en la tabla de hechos de alumnos no pueden analizarse desde la dimensión de docente, al igual que las que no se encuentran en la tabla de hechos de docente no pueden analizarse desde la dimensión de alumno. Esto se debe a que la clave unívoca de la dimensión desde la que no se puede analizar la medida no forma parte de la tabla de hechos.

DIMENSIÓN (elemento /total)		**MEDI DA**								
		(1)	**(2)**	**(3)**	**(4)**	**(5)**	**(6)**	**(7)**	**(8)**	**(9)**
TIEMPO	2º cuatrimestre (2011/12)	C	C	C	C	C	C	C	C	C
	Total	C	C	C	C	C	C	C	C	C
UNIVERSIDAD	Escuela politécnica (Leganés-UC3M)	C	C	C	C	C	C	C	C	C
	Total	C	C	C	C	C	C	C	C	C
TITULACIÓN	Com. en la Empresa (Periodismo)	C	C	C	C	C	C	C	C	C
	Total	C	C	C	C	C	C	C	C	C
TIPO ASIGNATURA	Prácticas Externas	C	C	C	C	C	C	C	C	C
	Total	C	C	C	C	C	C	C	C	C
CURSO	QUINTO	C	C	C	C	C	C	C	C	C
	Total	C	C	C	C	C	C	C	C	C
DEPARTAMENTO	Eléctrica (Eléctrica)	C	C	C	C	C	C	C	C	C
	Total	C	C	C	C	C	C	C	C	C
TURNO	MAÑANA	C	C	C	C	C	C	C	C	C
	Total	C	C	C	C	C	C	C	C	C
CONVOCATORIA	2	C	C	C	C	C	C	C	C	C
	Total	C	C	C	C	C	C	C	C	C

TIPO ACCESO UNIVERSIDAD	MAYORES DE 25	C	C	C	C	C	C	C	C	C
	Total	C	C	C	C	C	C	C	C	C
CONVOCATORIA ACCESO	EXTRAORDINARIA	C	C	C	C	C	C	C	C	C
	Total	C	C	C	C	C	C	C	C	C
CATEGORIA LABORAL	AYUDANTE DOCTOR	C	C	C	C	C	C	C	C	C
	Total	C	C	C	C	C	C	C	C	C
SEXO ALUMNO	HOMBRE	C	C	C	C	C	C	C	C	C
	Total	C	C	C	C	C	C	C	C	C
SEXO DOCENTE	MUJER	C	C	C	C	C	C	C	C	C
	Total	C	C	C	C	C	C	C	C	C
ESTADO TITULACIÓN	VIGENTE	C	C	C	C	C	C	C	C	C
	Total	C	C	C	C	C	C	C	C	C
ALUMNO	45568738T	C	C	C	C	C	C	n.a.	n.a.	n.a
	Total	C	C	C	C	C	C	n.a.	n.a.	n.a
DOCENTE	45570683U	n.a.	n.a.	n.a.	n.a.	n.a.	n.a.	C	C	C
	Total	n.a.	n.a.	n.a.	n.a.	n.a.	n.a.	C	C	C

Tabla 30. Validación datos

(1) No. Alumnos; **(2)** Nota Media Acceso a la Universidad; **(3)** Nota Media; **(4)** Edad Media Alumno; **(5)** Media de Dispensas; **(6)** Costes Tasas; **(7)** No. Docentes; **(8)** Nota Media Docente; **(9)** Edad Media Docente

Se concluye la correcta transformación del modelo, con una excepción a la *primera propiedad de Madhavan* (véase 2.5 Conversión entre modelos), ya que a consultas como ¿cuántos alumnos tiene un profesor? , ¿cuál es la nota media de los alumnos de un profesor? o ¿cuántos profesores tiene un alumno? no se puede responder en el *esquema en estrella* y si en el *E/R*.

Esto ocurre porque la dimensión de Alumno y Docente fue necesaria para el cálculo de las notas medias y números de alumnos y profesores. Sin embargo, bajar a este detalle obligó crear dos *tablas de hechos* perdiendo en la tabla de alumnos al identificador del docente y al revés, imposibilitando el cálculo de ciertas *medidas*.

Los modelos dimensionales son propios de otros dominios donde la herramienta de análisis no suele bajar a niveles de personas, pueden modelarse con una *tabla de hechos* o bien se opta por construir diferentes cubos (cada uno con un esquema en estrella

propio que analice *medidas* diferentes). Pero en el dominio de las universidades, tratado en este proyecto, este nivel tan bajo si resulta de interés. Así, existe un conflicto que lleva a no cumplir completamente la primera propiedad de conversión entre esquemas.

A modo de ejemplo se muestran los scripts de las consultas realizadas sobre el esquema *E/R* para las comprobaciones de la medida número de alumnos en las diferentes dimensiones:

SCRIP comprobación medida no. de alumno por dimensiones

```
-- No. de alumnos: TIEMPO
select count (distinct dni) from cursa
select count (distinct dni) from cursa where anio_academico=201112 and cuatrimestre =2

-- No. de alumnos: UNIVERSIDAD
select count (distinct c.dni) from cursa c, matricula m
select count (distinct c.dni) from cursa c, matricula m where c.dni=m.dni and id_centro=1

-- No. de alumnos: TITULACIÓN
select count (distinct dni) from cursa
select count (distinct dni) from cursa where id_asignatura=948

-- No. de alumnos: TIPO ASIGNATURA
select count (distinct c.dni) from cursa c
select count (distinct c.dni) from cursa c, asignatura a where c.id_asignatura=a.id_asignatura and
a.id_tipo_asignatura=7

-- No. de alumnos: CURSO
select count (distinct c.dni) from cursa c
select count (distinct c.dni) from cursa c, asignatura a where c.id_asignatura=a.id_asignatura and
curso=5

-- No. de alumnos: DEPARTAMENTO
select count (distinct c.dni) from cursa c
select count (distinct c.dni) from cursa c, asignatura a where c.id_asignatura=a.id_asignatura and
a.id_departamento=19

-- No. de alumnos: TURNO
select count (distinct dni) from cursa
select count (distinct dni) from cursa where turno='MAÑANA'

-- No. de alumnos: CONVOCATORIA
select count (distinct dni) from cursa
select count (distinct dni) from cursa where num_convocatoria=2

-- No. de alumnos: TIPO ACCESO UNIVERSIDAD
select count (distinct c.dni) from cursa c
select count (distinct c.dni) from cursa c, matricula m where c.dni=m.dni and tipo_acceso='MAYORES
DE 25'
-- No. de alumnos: CONVOCATORIA ACCESO
```

```
select count (distinct c.dni) from cursa c
select count (distinct c.dni) from cursa c, matricula m where c.dni=m.dni and convocatoria=2

-- No. de alumnos: CATEGORIA LABORAL
select count (distinct dni) from cursa
select count (distinct c.dni) from cursa c, imparte i where c.id_asignatura=i.id_asignatura and
c.anio_academico=i.anio_academico and i.dni in (select distinct d.dni from docente d where
d.id_cat_laboral=3)

-- No. de alumnos: SEXO ALUMNO
select count (distinct dni) from cursa
select count (distinct dni) from cursa where dni in (select distinct p.dni from persona p where
p.tipo='ALUMNO' and p.sexo='HOMBRE')

-- No. de alumnos: SEXO DOCENTE
select count (distinct dni) from cursa
select count (distinct c.dni) from cursa c, imparte i where c.id_asignatura=i.id_asignatura and
c.anio_academico=i.anio_academico and i.dni in (select distinct p.dni from persona p where
p.tipo='DOCENTE' and p.sexo='MUJER')

-- No. de alumnos: ESTADO TITULACION
select count (distinct c.dni) from cursa c
select count (distinct c.dni) from cursa c, matricula m, titulacion t where c.dni=m.dni and
m.id_titulacion=t.id_titulacion and id_estado=1

-- No. de alumnos: ALUMNO
select count (distinct dni) from cursa
select count (distinct dni) from cursa where dni='45568738T'
```

Tabla 31. SCRIP comprobación medida no. de alumno por dimensiones

La siguiente Tabla 32 se muestran tres consultas adicionales establecidas sobre los esquemas para comparar los resultados cruzando dimensiones, con el ffin de detectar posibles errores en los mecanismos de cruce del *cubo*. La columna (1) es el identificador de la comprobación llevada a cabo, las columnas (2) y (3) indican si hay diferencia en los datos y si ha llegado a una correcta transformación en el almacenamiento de la información, respectivamente:

(1)	CONSULTAS SOBRE LAS TABLASEN EL MODELO E/R	(2)	(3)
1	La nota media obtenida y número de alumnos en los diferentes cuatrimestres de los años académicos. *Con esta consulta comprobamos dos medidas cruzadas con una única dimensión desglosada en sus dos niveles.*	NO	SI

2	La nota media obtenida y número de alumnos en los diferentes tipos de asignatura a examen de los años académicos. *Con esta consulta comprobamos dos medidas cruzadas con dos dimensiones la convocatoria desglosada en su único nivel y el tiempo desglosado hasta el nivel inferior.*	NO	SI
3	La nota media obtenida y número de alumnos en las diferentes convocatorias a examen de los años académicos.	NO	SI

Tabla 32. Validación conversión entre modelos EER

Las querys que efectúan las consultas se detallan a continuación identificadas por la columna (1) de la Tabla 32. En la misma fila los elementos utilizados en el cubo:

(1)	QUERY	ELEMENTOS DE UNICUBE
1	**SELECT** ANIO_ACADEMICO,CUATRIMESTRE, AVG(NOTA), COUNT (DISTINCT (DNI)) **FROM** CURSA **GROUP BY** ANIO_ACADEMICO, CUATRIMESTRE **ORDER BY** 1, 2	**COLUMNAS:** Nota Media y No. Alumno **FILAS:** Dimensión TIEMPO expandida
2	**SELECT** ANIO_ACADEMICO, CUATRIMESTRE, NUM_CONVOCATORIA, AVG(NOTA), COUNT (DISTINCT (DNI)) **FROM** CURSA **GROUP BY** ANIO_ACADEMICO, CUATRIMESTRE, NUM_CONVOCATORIA **ORDER BY** 1, 2, 3	**COLUMNAS:** Nota Media y No. Alumno **FILAS:** Dimensión TIEMPO expandida y anidada la dimensión CONVOCATORIA
2	**SELECT** ANIO_ACADEMICO, CUATRIMESTRE, NUM_CONVOCATORIA, AVG(NOTA), COUNT (DISTINCT (DNI)) **FROM** CURSA **GROUP BY** ANIO_ACADEMICO, CUATRIMESTRE, NUM_CONVOCATORIA **ORDER BY** 1, 2, 3	**COLUMNAS:** Nota Media y No. Alumno **FILAS:** Dimensión TIEMPO expandida y anidada la dimensión CONVOCATORIA

Tabla 33. Consultas validación conversión

Todas las comprobaciones se realizan sobre un número reducido de registros, dado que el tratamiento de miles de tuplas en un GBBDDR es muy lento y las comprobaciones no se pueden realizar.

Una comprobación de la definición de las *tablas de hechos* consiste en verificar la clave primaria. Esta debe ser la concatenación de las claves ajenas que relacionan estas tablas con las dimensiones. Se puede comprobar esta unicidad de la clave primaria con una consulta que agrupe estos campos, y devuelva sólo los que se repiten, si no se obtiene resultado la tabla cumple el criterio de clave única.

SCRIP comprobación unicidad de clave en HECHOS_ALUMNO
-- Comprobación de clave de la tabla de HECHOS_ALUMNO, el resultado debe ser 0 registros SELECT COUNT(*) FROM HECHOS_ALUMNO GROUP BY SK_ALUMNO, SK_TIEMPO, SK_CENTRO, SK_ASIGNATURA, SK_TIPO_ASIGNATURA,SK_CURSO, SK_DEPARTAMENTO, SK_TURNO, SK_CONVOCATORIA, SK_TIPO_ACCESO, SK_CONVOCATORIA_ACCESO, SK_CAT_LABORAL, SK_SEXO_ALUMNO, SK_SEXO_DOCENTE, SK_ESTADO HAVING (COUNT(*)>1)

Tabla 34. SCRIP comprobación unicidad de clave en HECHOS_ALUMNO

SCRIP comprobación unicidad de clave en HECHOS_DOCENTE
-- Comprobación de clave de la tabla de HECHOS_DOCENTE, el resultado debe ser 0 registros SELECT COUNT(*) FROM HECHOS_DOCENTE GROUP BY SK_DOCENTE, SK_TIEMPO, SK_CENTRO, SK_ASIGNATURA, SK_TIPO_ASIGNATURA,SK_CURSO, SK_DEPARTAMENTO, SK_TURNO, SK_CONVOCATORIA, SK_TIPO_ACCESO, SK_CONVOCATORIA_ACCESO, SK_CAT_LABORAL, SK_SEXO_ALUMNO, SK_SEXO_DOCENTE, SK_ESTADO HAVING (COUNT(*)>1)

Tabla 35. SCRIP comprobación unicidad de clave en HECHOS_DOCENTE

La realización de estas pruebas es positiva en ambos casos.

Capítulo 7

En este capítulo se exponen las conclusiones más importantes extraídas de la realización del presente proyecto final de carrera.

7. CONCLUSIONES

Tras el desarrollo de este proyecto se puede concluir que la construcción de un sistema de *BI* no es un proceso trivial. Por norma, la información está contenida en estructuras que no se adaptan al análisis dimensional. Los modelos de datos deben ser adaptados para adecuarse a esquemas que soporten datos dimensionales.

El correcto diseño de un modelo, facilita el tratamiento de la información en sistemas como los *cubos*, que pueden operar rápidamente sobre gran cantidad de datos, generando análisis en tiempo real que de otra forma, costarían tiempo y recursos, como se ha demostrado en la fase de validación del sistema.

Pero las conversiones entre modelos no siempre obedecen a reglas establecidas, y el diseño de *esquemas en estrella* o en *copo de nieve*, basados en otros modelos propios de las *BBDDR*, aún no se encuentra automatizado y se requiere de un gran aporte humano que complemente a los sistemas de inteligencia artificial o basados en conocimiento, capaces de aproximar estas conversiones.

A nivel económico, una transformación optimizada, que opere con bastas cantidades de datos, mucho mayores a las manejadas en el desarrollo de este proyecto formativo, requiere de herramientas costosas y expertos que las manejen.

Además de ser necesario un proceso de mantenimiento del modelo dimensional, ya que la información se transforma y cambia de modelos con una *ETL* hoy, pero mañana, seguirá almacenándose en sistemas heterogéneos que habrá que volver a convertir y que pueden evolucionar, provocando modificaciones en la *ETL* que convierte los datos de un modelo a otro.

La información es poder, y la construcción de un *DWH* no supone un simple cambio en el almacenamiento de la información. La decisión de implementar un almacén de datos de esas características, implica una integración de todos los datos, contenida en diversos sistemas operacionales, en un sistema global. Posibilitando una explotación de datos que aporte mucha y muy valiosa información.

A nivel empresarial, el correcto diseño de los *Datamarts* que conforman un *DWH* ofrece una ventaja competitiva a la organización, puesto que puede optar por usar potentes herramientas de ayuda a toma de decisiones, como los *CM* y los análisis *OLAP*, que la posicionen líder en el mercado.

Capítulo 8

Esta sección versará sobre las posibles evoluciones del proyecto actual.

8. LÍNEAS FUTURAS

Este proyecto puede ampliar muchos aspectos diferentes:

- Los almacenes de datos son un tema muy amplio y su estudio puede abarcar más aspectos de los tratados en este documento. Los motores de bases de datos enfocados a los *DWH* y a modelos que soporten el almacenamiento de datos dimensionales pueden resultar interesantes.

- Un sistema de *BI* puede ofrecer diversos productos, no solo los *cubos multidimensionales*, como los *CM* o las herramientas de apoyo a la toma de decisiones, tocados muy superficialmente en este desarrollo. El análisis e implementación de estos productos es de gran interés empresarial.

- En cuanto a la conversión entre modelos de datos, la automatización del proceso presenta una rama de investigación muy compleja y prometedora. Además, se puede analizar este punto con herramientas comerciales diferentes, o incluso, se podría enfocar hacia el diseño de una herramienta nueva, capaz de competir con las actuales.

- En lo referente a la explotación de modelos de datos dimensionales, se ha desarrollado un *cubo*, pero un análisis completo de la información podría incluir informes profesionales, que podrían implementarse con componentes descritos en el proyecto (Report Studio, de IBM Cognos).

Bibliografía

[IBM, 2011] IBM: Business Analytics & Optimization. Accesible en http://www-05.ibm.com/services/es/bcs/html/bao/ . Consultada en octubre de 2011.

[Davenport, T.; Prusak, L., 1998] Davenport, T.; Prusak, L. 1998. Working Knowledge: How Organizations Manage What They Know. Harvard Business School Press.

[Gartner Research, 2005] Gartner Research. Accesible en http://www.gartner.com/technology/research.jsp. Consultada en octubre de 2011.

[Imnon, W.H., 1992] Imnon, W.H. 1992. Building the Data Warehouse. QED Technical Publishing Group.

[Kimbal R., 1998] Kimball, Ralph. 1998. The Data Warehouse Lifecycle Toolkit. Wiley.

[Turban E., 1995] Turban, E. 1995. Decision support and expert systems: management support systems. Englewood Cliffs, N.J., Prentice Hall.

[Castro Galán E., 2003] Castro Galán, Elena. 2003. Una aproximación a la reutilización de estructuras documentales definidas con XML en el marco WEB semántica. Universidad Carlos III de Madrid.

[Madhavan, et. al., 2002] Madhavan J. 2002. Representing and Reasoning about Mappings between Domains Models. American Association for Artificial Intelligence.

[Bernstein, A., Levy, A. & Pottinger, R., 2000] Bernstein, A., Levy, A. & Pottinger, R. 2000. A Vision for Management of Complex Models.

[Cedeño A., 2005] Cedeño Trujillo, A. 2005. Modelo multidimensional. Accesible en http://rii.cujae.edu.cu/index.php/revistaind/search/authors/view?firstName=Alexis&middleName=&lastName=Cede%C3%B1o%20Trujillo&affiliation=&country= . Consultada en abril de 2012.

[1] Sinnexus: Business Intelligence + Informática estratégica. Accesible en http://www.sinnexus.com/business_intelligence/index.aspx. Consultada en noviembre de 2011.

[2] Monografias: Inteligencia de Negocios. Accesible en http://www.monografias.com/trabajos75/inteligencia-negocios/inteligencia-negocios3.shtml. Consultada en noviembre de 2011.

[3] Microsoft: TechNet. Accesible en http://technet.microsoft.com/es-es/library/ms175367(SQL.90).aspx. Consultada en noviembre de 2011.

[4] Microsoft: TechNet. Accesible en http://technet.microsoft.com/es-es/library/ms174783(SQL.90).aspx. Consultada en noviembre de 2011.

[5] Gestión del conocimiento. Accesible en http://www.gestiondelconocimiento.com/conceptos_diferenciaentredato.htm. Consultada en octubre de 2011.

[6] DATA WAREHOUSING: Investigación y Sistematización de Conceptos. Accesible en http://es.scribd.com/doc/55234594/44/Esquema-Copo-de-Nieve. Consultada en noviembre de 2011.

ANEXO A - Manual de Usuario Analysis Studio

A.1. Objetivo

Este anexo sirve de guía para la utilización y máximo aprovechamiento del cubo multidimensional UNICUBE, facilitando así el análisis de los datos en él contenidos.

Para ello, en primer lugar, se describen los elementos básicos necesarios para realizar un análisis (objetos insertables y espacio de trabajo). Esta primera parte tiene como fin que el usuario se familiarice con el entorno de trabajo.

A continuación se describen los menús/comandos disponibles y las acciones más habituales que el usuario podrá realizar: insertar filas, columnas, aplicar filtros a los datos, etc.

Una vez introducidos todos estos conceptos, se simulan unos análisis, con ejemplos concretos, con el fin de obtener una mayor comprensión de lo anteriormente expuesto.

A.2. Alcance

Este documento muestra las funcionalidades del cubo UNICUBE.

Se detallan las opciones disponibles para el usuario así como unos casos prácticos que, a modo de ejemplo, simulan un análisis para mostrar cómo se pueden aplicar en la práctica las funcionalidades explicadas. Los casos prácticos han sido divididos en niveles de análisis (básico, medio y avanzado) teniendo en cuenta que los usuarios tienen necesidades distintas.

A.3. Acceso al Portal de Cognos y al cubo

El acceso al cubo se realiza a través del Portal de Cognos, ubicado en la máquina local:

http://localhost/cognos8

Cuando se dota de seguridad al Portal de Cognos, es preciso identificarse con login y contraseña, en la imagen se muestra un ejemplo en que se loga contra el LDAP corporativo de una compañía.

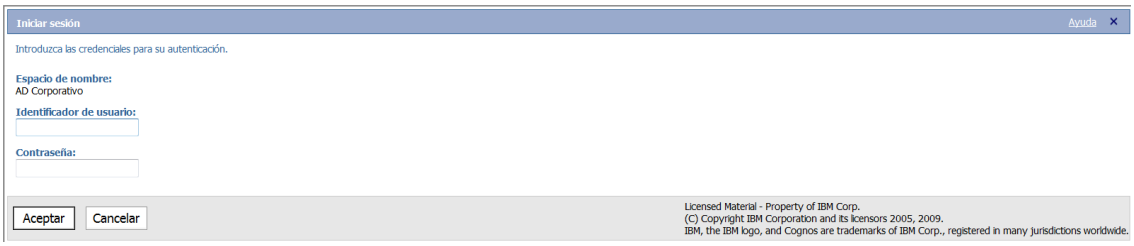


Ilustración 50. Inicio sesión Cognos con identificación

Se accede a la página principal:

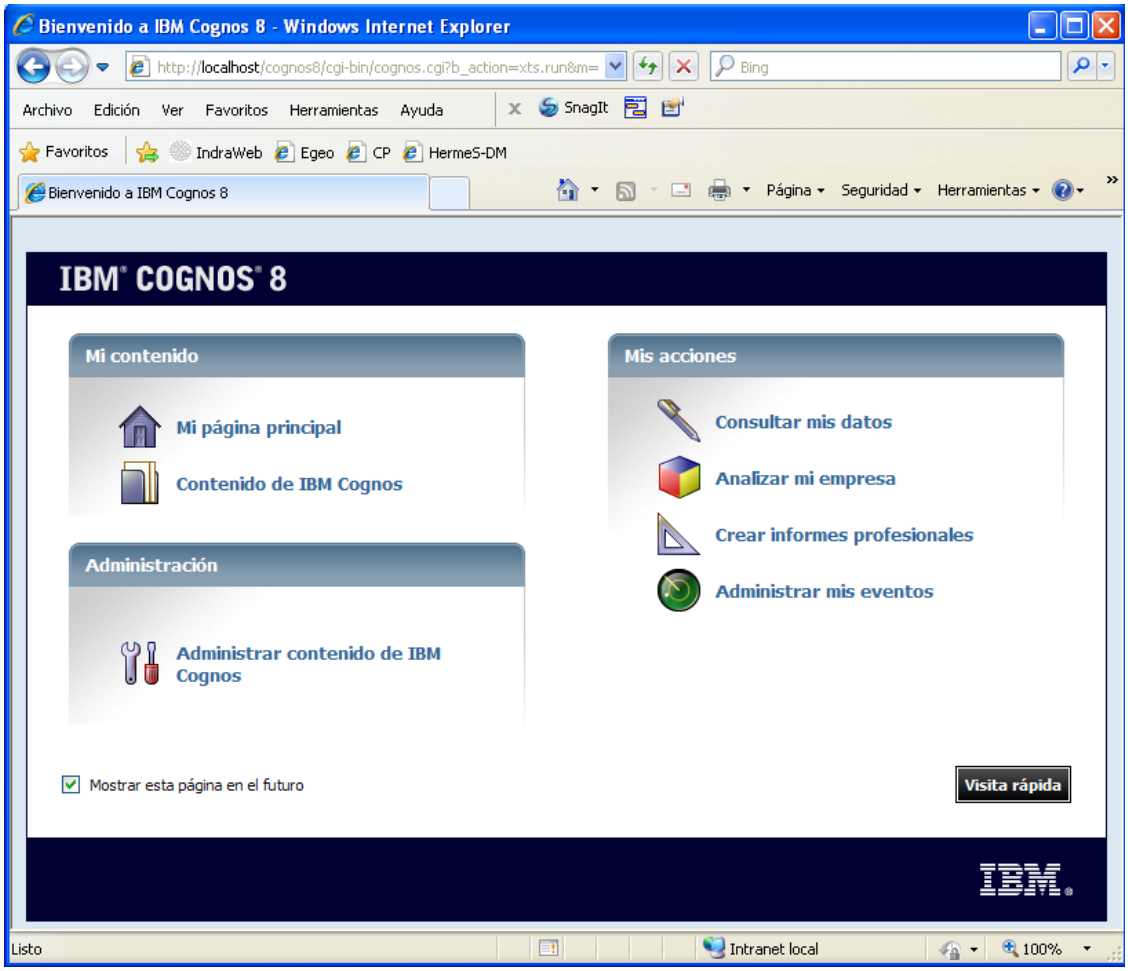


Ilustración 51. Página inicio Cognos

Pulsar la opción Analizar mi empresa. Seleccionar el modelo de datos correspondiente a UNICUBE.

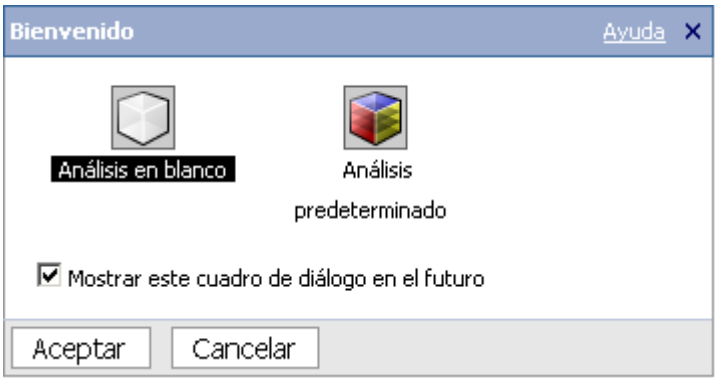


Ilustración 52. Analizar Mi Empresa

Se mostrará la opción de comenzar un análisis nuevo (desde cero) o visualizar un análisis predeterminado.

Seleccionar la opción que deseada y pulsar Aceptar.

Eligiendo la opción Análisis en blanco se muestra el espacio de trabajo sobre el cual se puede realizar un análisis.

Si se desea elegir la opción Análisis predeterminado, se abrirá el espacio de trabajo con una tabla cruzada formada por las dos primeras dimensiones y el primer indicador o medida.

A.4. Análisis del Cubo UNICUBE

El análisis se estructura en dimensiones, las cuales se podrán incluir según interese visualizar los datos, permitiendo realizar un análisis multidimensional.

- En el panel de la izquierda aparecen los objetos insertables. Los objetos insertables son los elementos que es posible seleccionar para realizar el análisis deseado.

- En el panel de la derecha se encuentra el espacio de trabajo. En este espacio de trabajo se arrastran los objetos insertables para realizar un análisis.

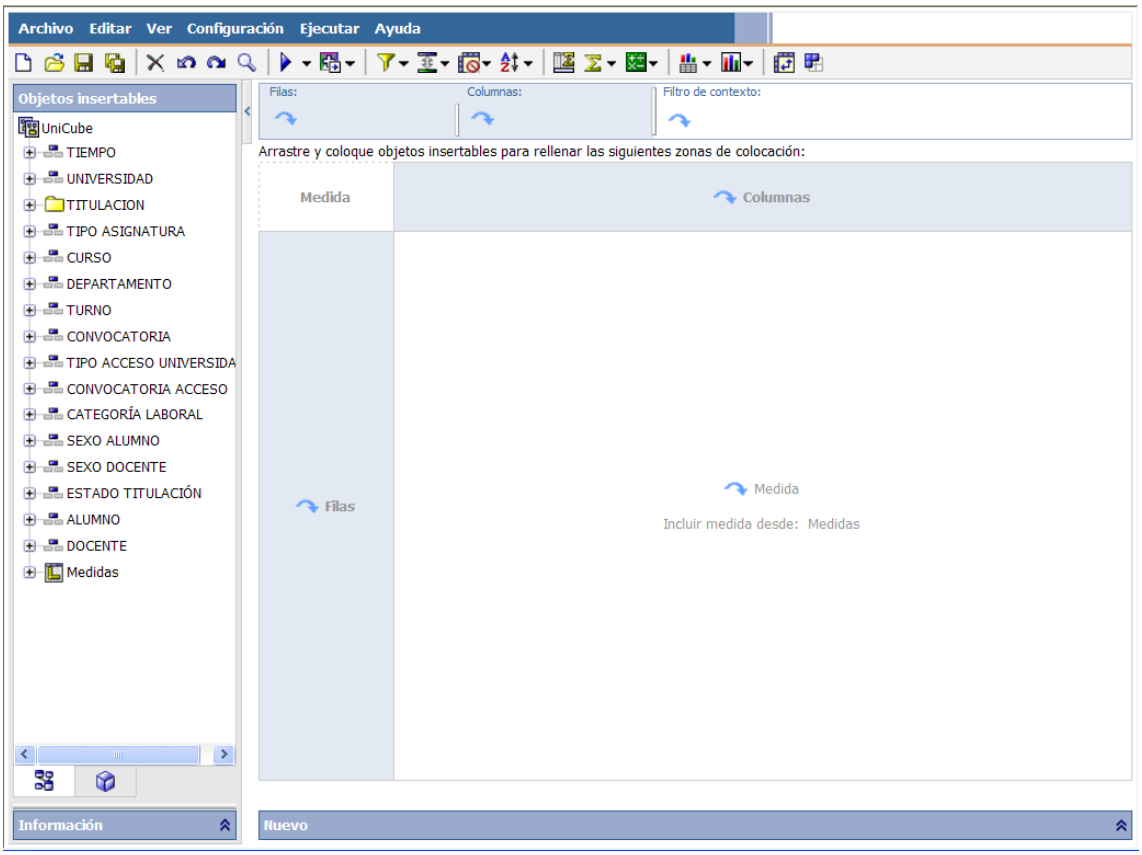


Ilustración 53. Análisis Cubo UNICUBE

Existen dos tipos de objetos insertables:

- Dimensiones
- Atributos de hechos, indicadores o medidas

En el panel de la izquierda, donde figuran los objetos insertables, está la información (input) que es posible añadir o utilizar como filtro en el análisis. El panel de la derecha es en el que se visualizan (output) las acciones que van realizándose y donde aparece el resultado en cada momento.

A.5. Guía para realizar un análisis

Una gran mayoría de las acciones que se quiere realizar, es posible hacerlas arrastrando objetos insertables al espacio de trabajo y con barra de herramientas superior. A continuación, se describe la navegación más usual.

A continuación se expone una manera de realizar un análisis con el cubo multidimensional. Muchas de las acciones se pueden hacer de varias maneras.

Puede darse el caso de que se comentan fallos durante la navegación o que involuntariamente se realicen cruces que no tienen sentido. Se puede hacer uso del botón de Deshacer para volver a la vista anterior.

A.5.1. Slicing y Dicing. Insertar filas y columnas

Para insertar filas o columnas simplemente se debe buscar en el panel de la izquierda, pinchar en el objeto a insertar, arrastrarlo y soltarlo en Filas o Columnas. El resultado de arrastrar un elemento de una dimensión es que se generarán tantas filas/columnas como elementos existan en el nivel inmediatamente inferior.

El siguiente ejemplo muestra como arrastrando el elemento Grado en Ingeniería Eléctrica (Nivel Titulación), nos inserta como filas en el espacio de trabajo sus elementos hijos (Nivel Asignatura). Los indicadores que salgan de este análisis sólo aplicarán a la titulación Grado en Ingeniería Eléctrica.

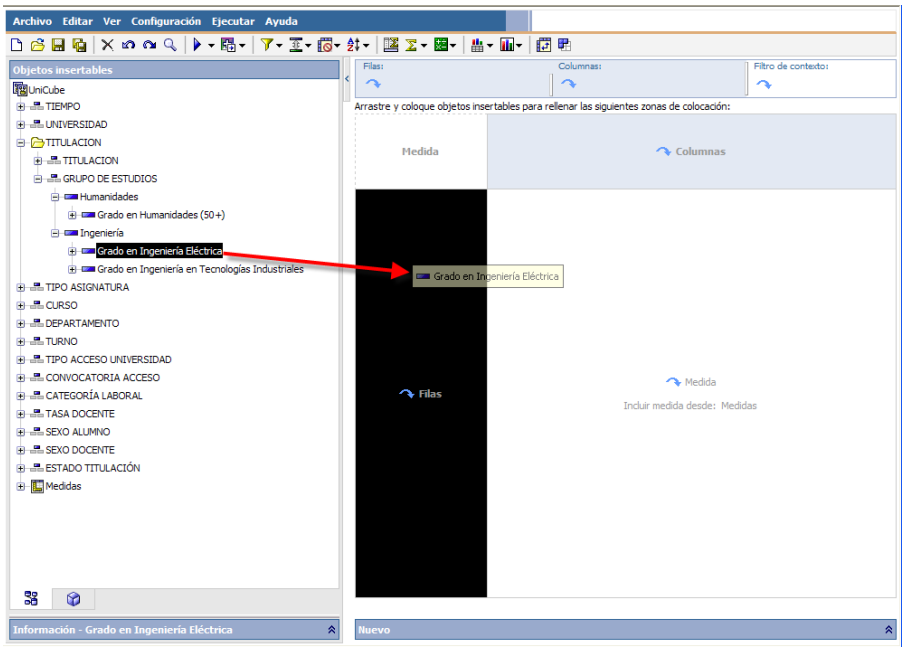


Ilustración 54. Inserción de fila en el cubo

El resultado será el siguiente:

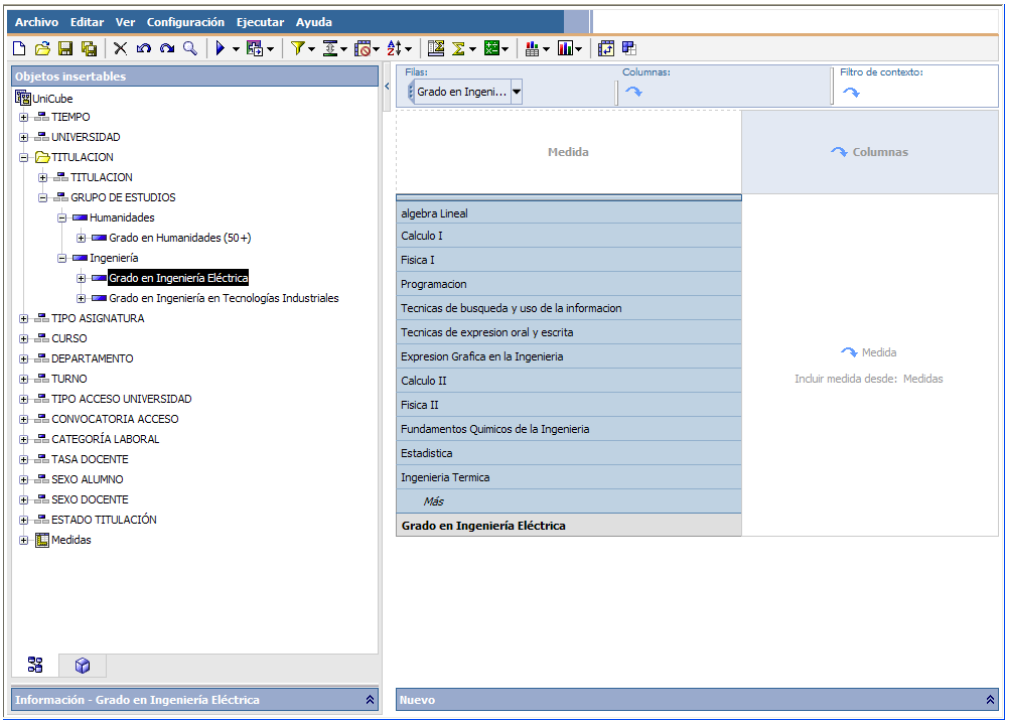


Ilustración 55. Resultado inserción de fila en el cubo

Pulsando el botón derecho del ratón sobre un elemento existe la opción de Insertar como filas/columnas (equivalente a arrastrar el elemento).

Si en lugar de arrastrar un elemento, se prefiere analizar todos los elementos de un nivel inferior de una dimensión, se utilizará la opción Insertar nivel. En este caso los indicadores aplicarán a toda la dimensión.

En el siguiente ejemplo, se selecciona el elemento “2003/04” y se selecciona Insertar nivel (año académico) por lo que aparecen todos los cuatrimestres.

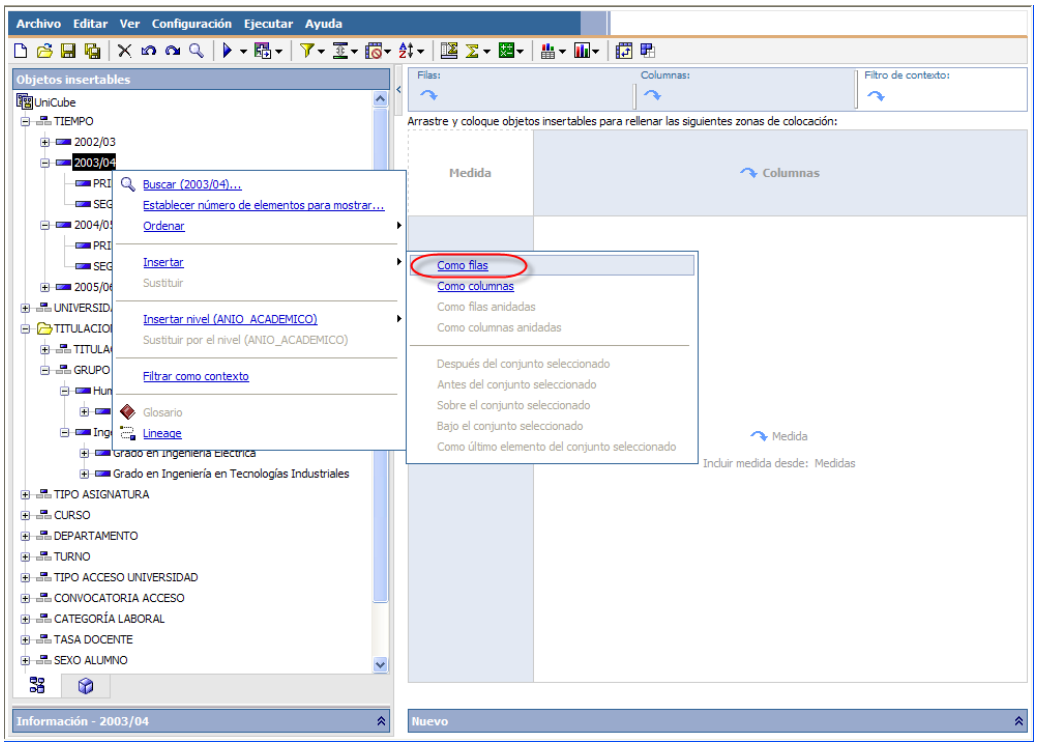


Ilustración 56. Insertar nivel como filas

El resultado es el siguiente:

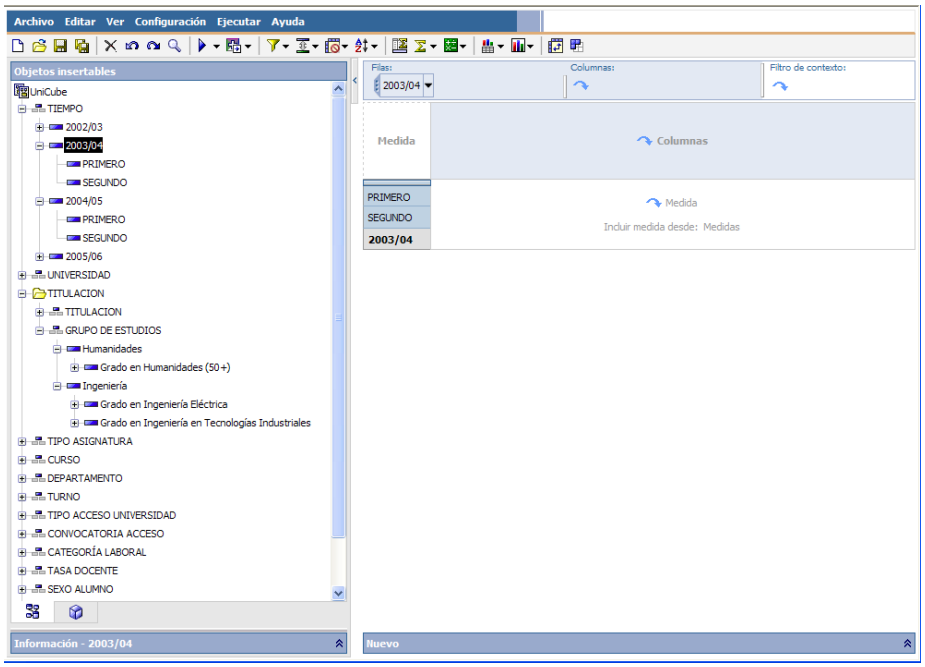


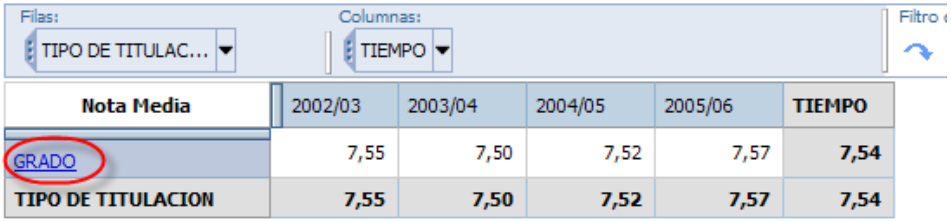
Ilustración 57. Resultado insertar nivel como filas

A.5.2. Drill-down. Exploración descendente

La exploración descendente de una dimensión consiste en ir bajando por la jerarquía de la dimensión hasta los niveles inferiores, de forma que se obtiene el detalle de los datos a partir de resúmenes de estos.

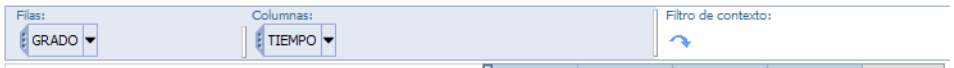
Para realizar drill-down en el cubo basta con hacer doble clic sobre el elemento de la dimensión que se desea explorar. El elemento puede encontrarse tanto en las filas como en las columnas.

Pulsando sobre el elemento Grado obtenemos el detalle de todos los hijos de Grado, las titulaciones:



Nota Media	2002/03	2003/04	2004/05	2005/06	TIEMPO
GRADO	7,55	7,50	7,52	7,57	7,54
TIPO DE TITULACION	7,55	7,50	7,52	7,57	7,54

Ilustración 58. Drill-down ejemplo 1



Nota Media	2002/03	2003/04	2004/05	2005/06	TIEMPO
Grado en Humanidades	7,56	7,50	7,50	7,56	7,53
Grado en Ingeniería Eléctrica	7,55	7,50	7,48	7,56	7,52
Grado en Ingeniería en Tecnologías Industriales			7,56	7,65	7,58
GRADO	7,55	7,50	7,52	7,57	7,54

Ilustración 59. Drill-down ejemplo 2

Se puede repetir la operación hasta llegar al nivel inferior de la dimensión, por ejemplo pulsando una titulación, se obtienen las asignaturas:

Filas: Grado en Ingeni... Columnas: TIEMPO Filtro de contexto:

Nota Media	2002/03	2003/04	2004/05	2005/06	TIEMPO
algebra Lineal	7,59				7,59
Calculo I	7,52				7,52
Fisica I	7,57				7,57
Programacion	7,75				7,75
Tecnicas de busqueda y uso de la informacion	7,49				7,49
Tecnicas de expresion oral y escrita	7,56				7,56
Expresion Grafica en la Ingenieria	7,59				7,59
Calculo II	7,39				7,39
Fisica II	7,44				7,44
Fundamentos Quimicos de la Ingenieria	7,62				7,62
Estadistica	7,52				7,52
Ingenieria Termica		7,40			7,40
Más					
Grado en Ingeniería Eléctrica	**7,55**	**7,50**	**7,48**	**7,56**	**7,52**

Ilustración 60. Drill-down ejemplo 3

Una alternativa que permite profundizar en todos los niveles a la vez, sin necesidad de seleccionar uno concreto, es pulsar la opción Bajar un nivel. La opción se encuentra pulsando sobre la dimensión completa y no sobre uno de sus elementos con el botón derecho:

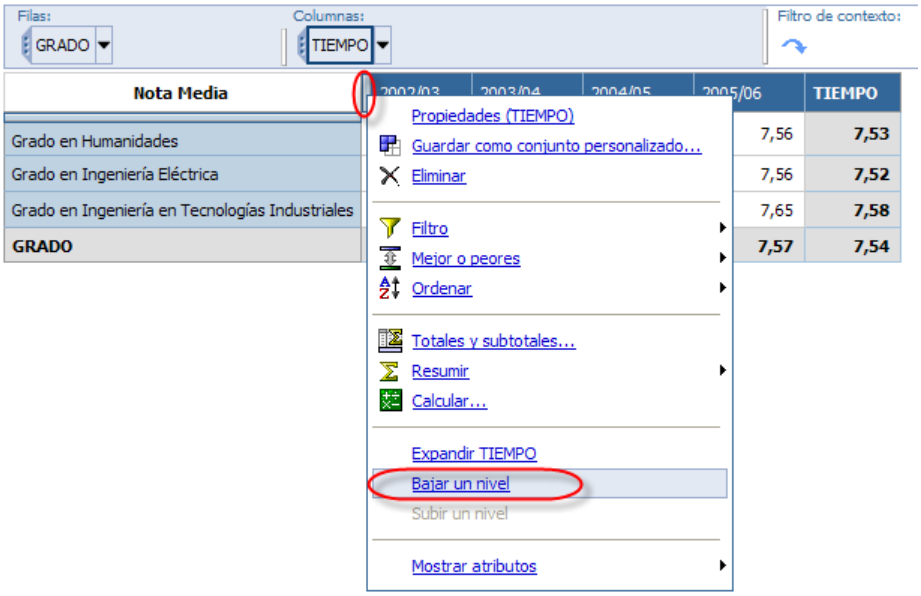


Ilustración 61. Opción Bajar un nivel

Se obtiene el siguiente resultado:

Filas: GRADO | Columnas: TIEMPO (profund... | Filtro de contexto:

Nota Media	PRIMERO	SEGUNDO	PRIMERO	SEGUNDO	PRIMERO	SEGUNDO	PRIMERO	SEGUN
Grado en Humanidades	7,49	7,61	7,52	7,48	7,53	7,48	7,58	
Grado en Ingeniería Eléctrica	7,58	7,51	7,46	7,53	7,46	7,49	7,63	
Grado en Ingeniería en Tecnologías Industriales					7,52	7,57	7,64	
GRADO	**7,54**	**7,56**	**7,49**	**7,50**	**7,51**	**7,52**	**7,61**	

Ilustración 62. Resultado bajar un nivel

Sin embargo esta alternativa pierde la visión del nivel superior y los datos pueden resultar confusos. Si no se desea perder los niveles superiores, pulsar Expandir nivel.

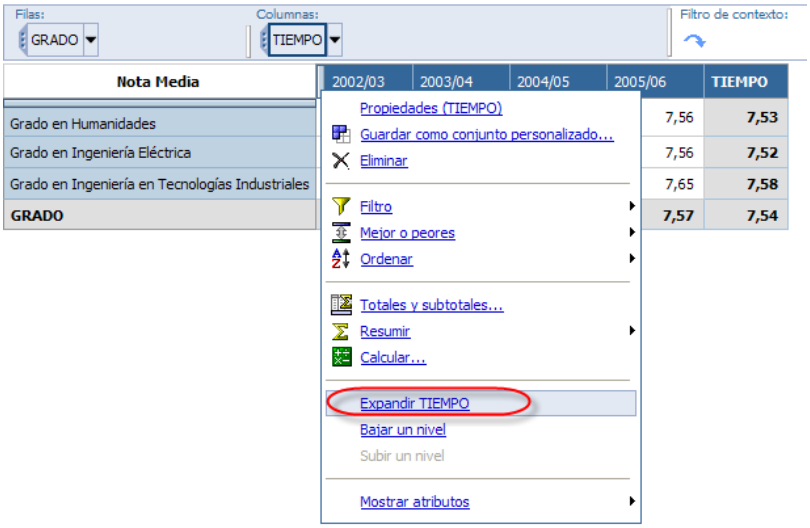


Ilustración 63. Opción Expandir nivel

El resultado obtenido es más claro:

Filas: GRADO | Columnas: Combinado | Filtro de contexto:

Nota Media	2002/03			2003/04			2
	PRIMERO	SEGUNDO	**2002/03**	PRIMERO	SEGUNDO	**2003/04**	
Grado en Humanidades	7,49	7,61	**7,56**	7,52	7,48	**7,50**	
Grado en Ingeniería Eléctrica	7,58	7,51	**7,55**	7,46	7,53	**7,50**	
Grado en Ingeniería en Tecnologías Industriales							
GRADO	**7,54**	**7,56**	**7,55**	**7,49**	**7,50**	**7,50**	

Ilustración 64. Resultado Expandir nivel

A.5.3. Roll-up. Exploración ascendente

La exploración ascendente de una dimensión consiste en ir remontando la jerarquía de la dimensión hasta los niveles superiores, de forma que se obtienen agrupaciones de los datos partiendo de un detalle.

Para realizar roll-up en el cubo basta con hacer doble clic sobre el total mostrado, correspondiente al elemento inferior de la dimensión a un nivel que no es el superior. El total puede corresponder tanto filas como a columnas.

Pulsando sobre el elemento Grado en Ingeniería Eléctrica obtenemos el agrupado de los hijos, las asignaturas:

Filas: Grado en Ingeni... Columnas: Combinado Filtro de cont

Nota Media	2002/03			2003/04	
	PRIMERO	SEGUNDO	**2002/03**	PRIMERO	SEGUN
algebra Lineal	7,59		**7,59**		
Calculo I	7,52		**7,52**		
Fisica I	7,57		**7,57**		
Programacion	7,75		**7,75**		
Tecnicas de busqueda y uso de la informacion	7,49		**7,49**		
Tecnicas de expresion oral y escrita	7,56		**7,56**		
Expresion Grafica en la Ingenieria		7,59	**7,59**		
Calculo II		7,39	**7,39**		
Fisica II		7,44	**7,44**		
Fundamentos Quimicos de la Ingenieria		7,62	**7,62**		
Estadistica		7,52	**7,52**		
Ingenieria Termica				7,40	
Más					
Grado en Ingeniería Eléctrica	**7,58**	**7,51**	**7,55**	**7,46**	

Ilustración 65. Roll-up ejemplo 1

Filas: GRADO Columnas: Combinado Filtro de

Nota Media	2002/03			2003/04	
	PRIMERO	SEGUNDO	**2002/03**	PRIMERO	SE
Grado en Humanidades	7,49	7,61	**7,56**	7,52	
Grado en Ingeniería Eléctrica	7,58	7,51	**7,55**	7,46	
Grado en Ingeniería en Tecnologías Industriales					
GRADO	**7,54**	**7,56**	**7,55**	**7,49**	

Ilustración 66. Roll-up ejemplo 2

La operación puede repetirse hasta llegar al nivel superior de la dimensión:

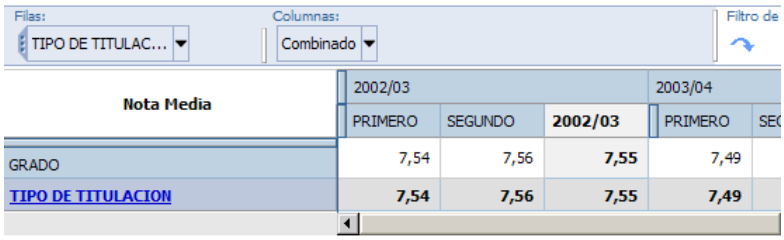


Ilustración 67. Resultado Roll-up

A.5.4. Insertar filtros

Para insertar filtros, seleccionar el objeto por el que queramos filtrar los datos mostrados y arrastrar hasta "Filtro de contexto", en el panel superior. Se pueden insertar varios filtros repitiendo esta misma operación.

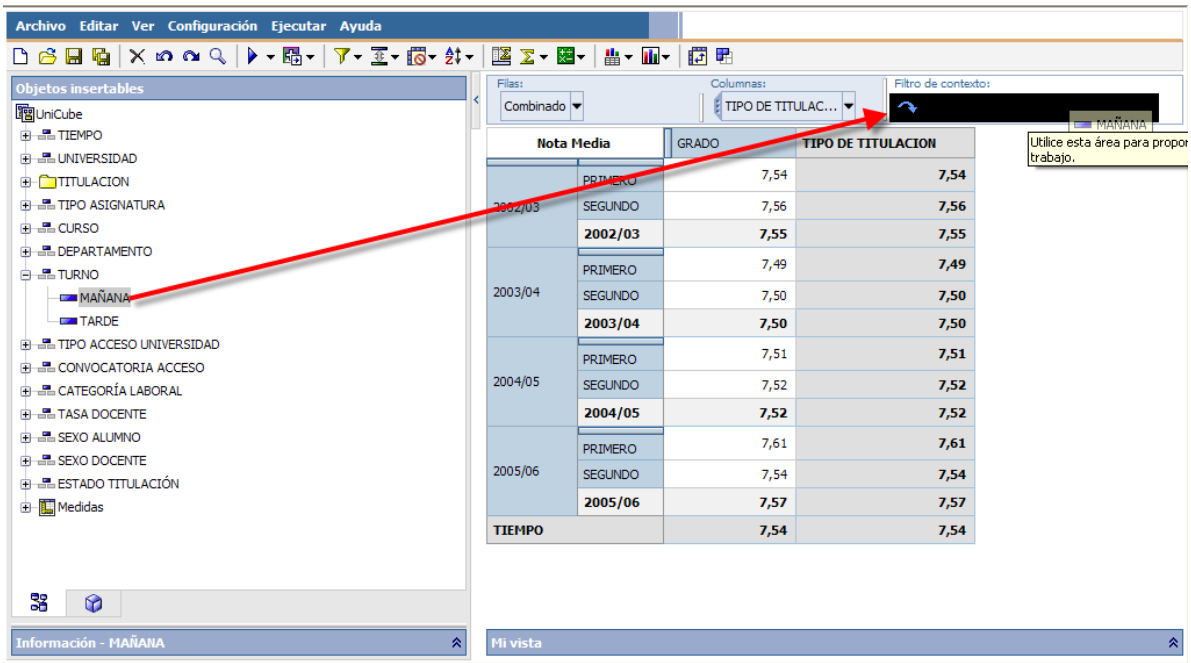


Ilustración 68. Inserción filtro de contexto

El resultado obtenido deja la estructura de la tabla intacta, no altera ni filas ni columnas. Sin embargo, los datos cambian, aplicándose el filtro.

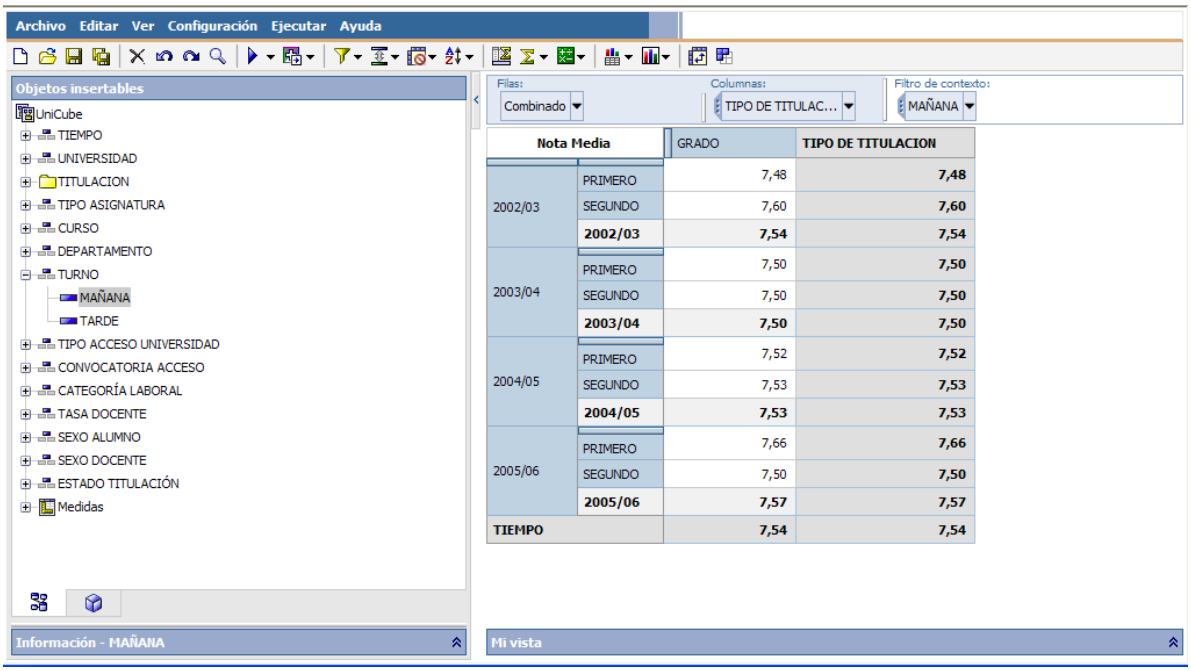


Ilustración 69. Resultado inserción filtro de contexto

Otra forma de filtrar es pulsando el botón derecho un elemento y seleccionando la opción Filtrar como contexto.

A.5.5. Insertar medidas

Para insertar medidas, seleccionar la medida o medidas que se desea visualizar y arrastrarlas hasta el área interior.

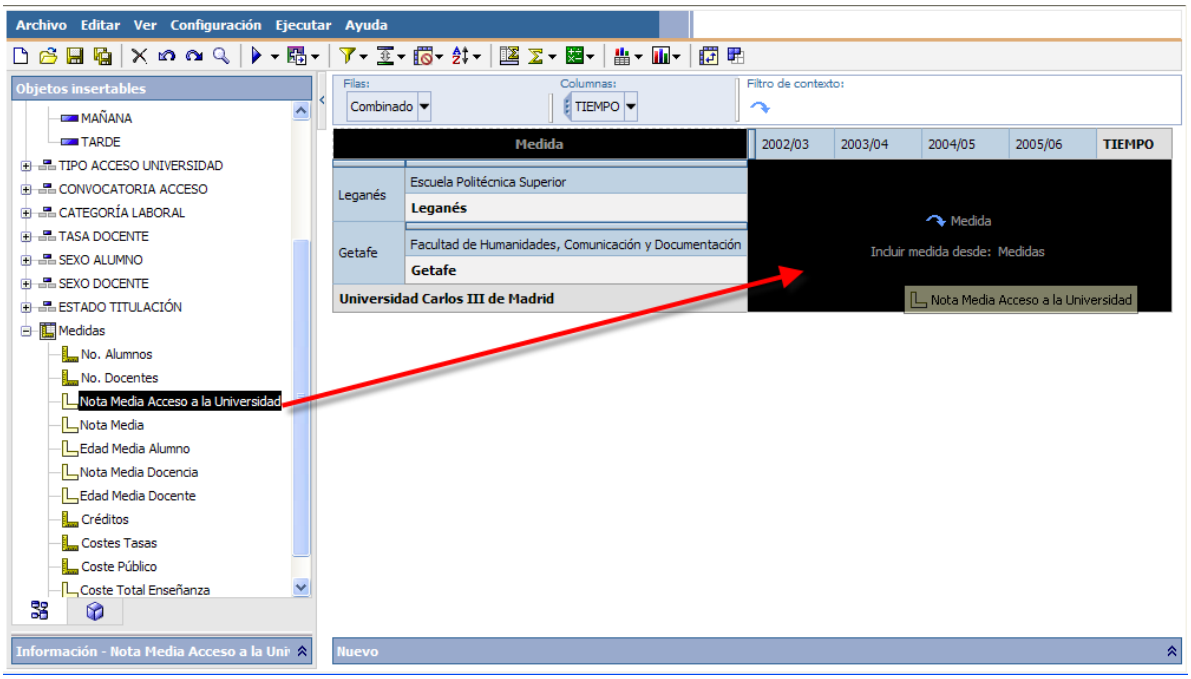


Ilustración 70. Insertar medidas

El resultado será el siguiente:

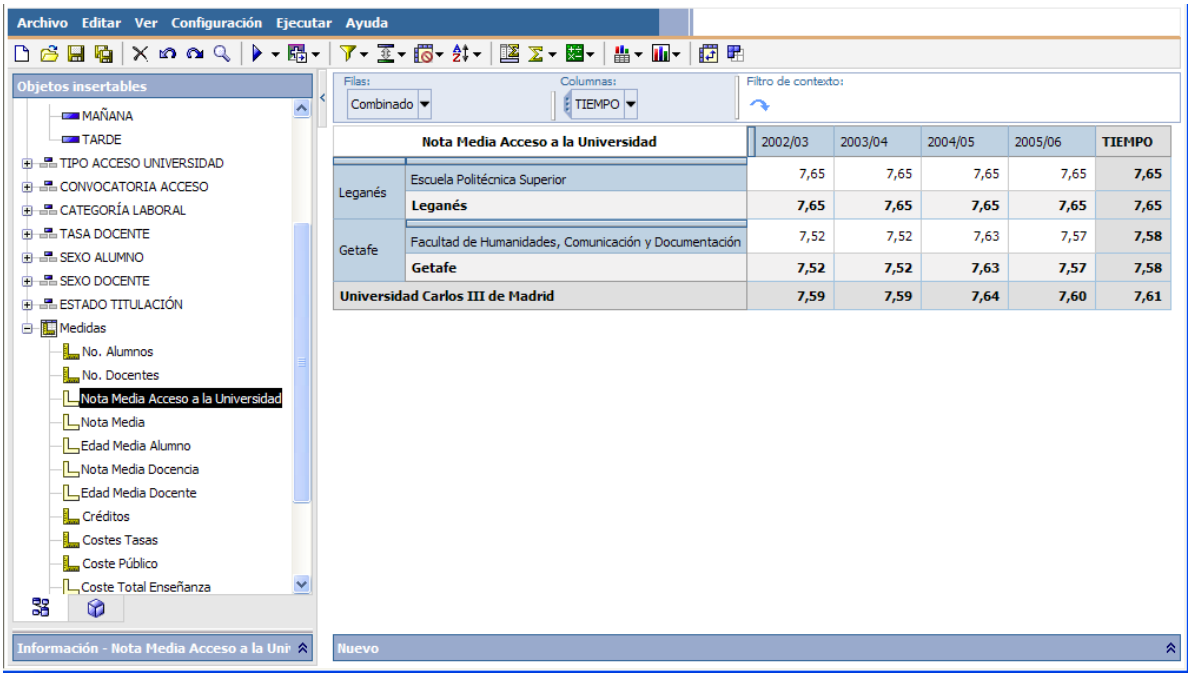


Ilustración 71. Resultado Insertar medidas

Para insertar varias medidas deben insertarse como filas o columnas. Las medidas se pueden seleccionar simultáneamente pulsado la tecla “Ctrl”.

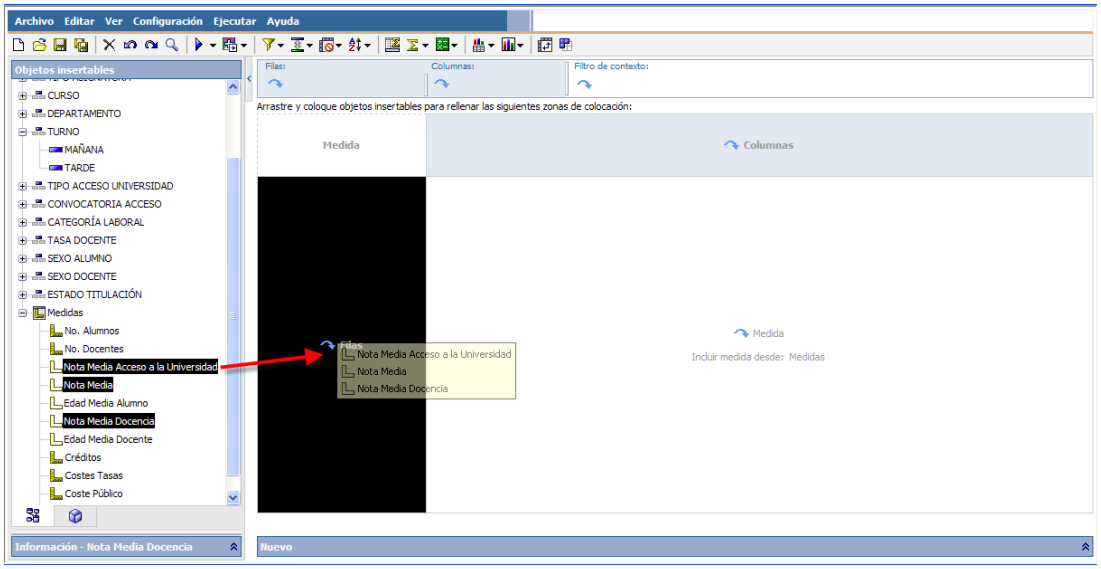


Ilustración 72. Insertar varias medidas

El resultado sería el siguiente:

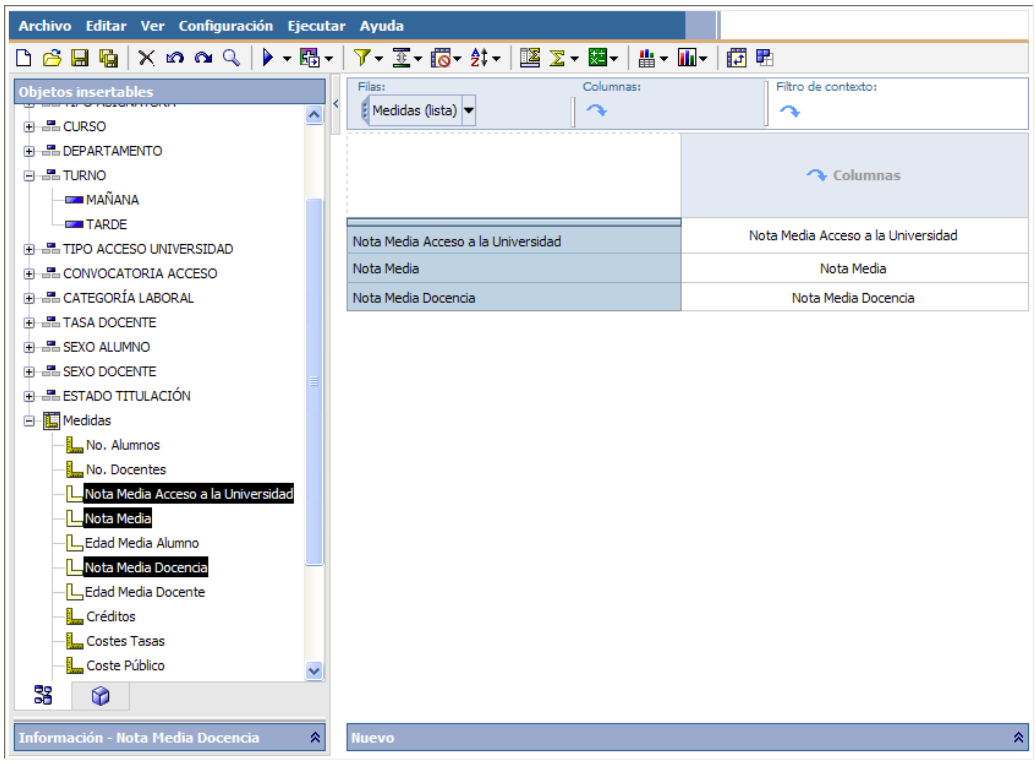


Ilustración 73. Resultado insertar varias medidas

A.5.6. Pivot. Intercambio de filas y columnas

En este ejemplo se visualizaría la información más fácilmente si las filas se intercambiasen con las columnas. Existe una opción para realizar este cambio. En la barra de herramientas superior pulsar el botón "Cambiar filas y columnas" .

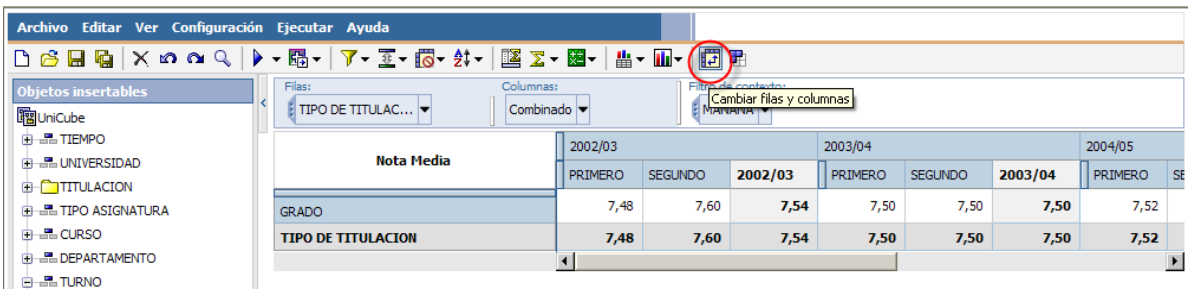


Ilustración 74. Cambiar filas y columnas

El resultado sería:

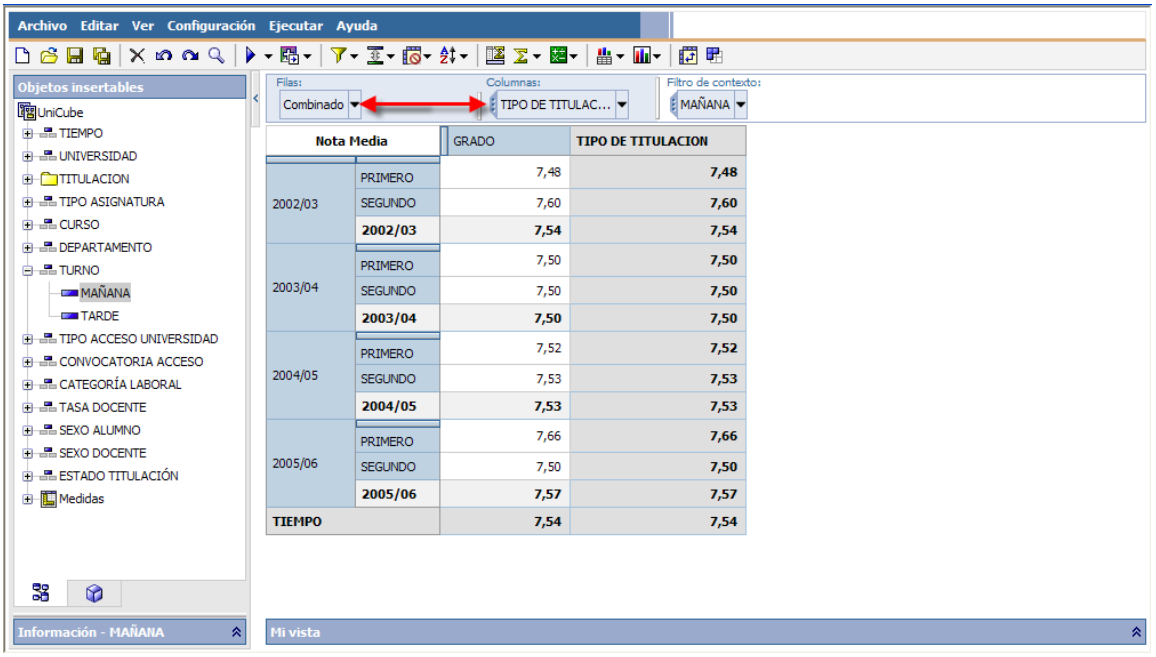


Ilustración 75. Resultado cambiar filas y columnas

A.5.7. Nesting. Anidamiento

Tanto con dimensiones como con medidas, se pueden realizar anidamientos sobre las filas o las columnas. Para ello, arrastrar el objeto hasta la zona en la que se desee anidar.

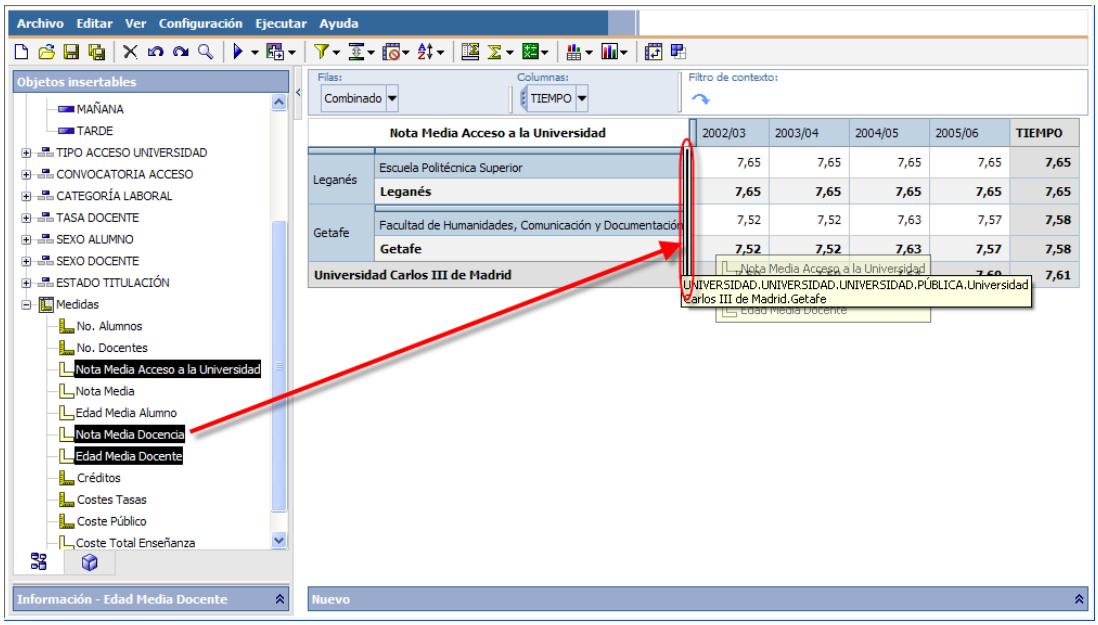


Ilustración 76. Nesting. Anidamiento

El resultado sería el siguiente:

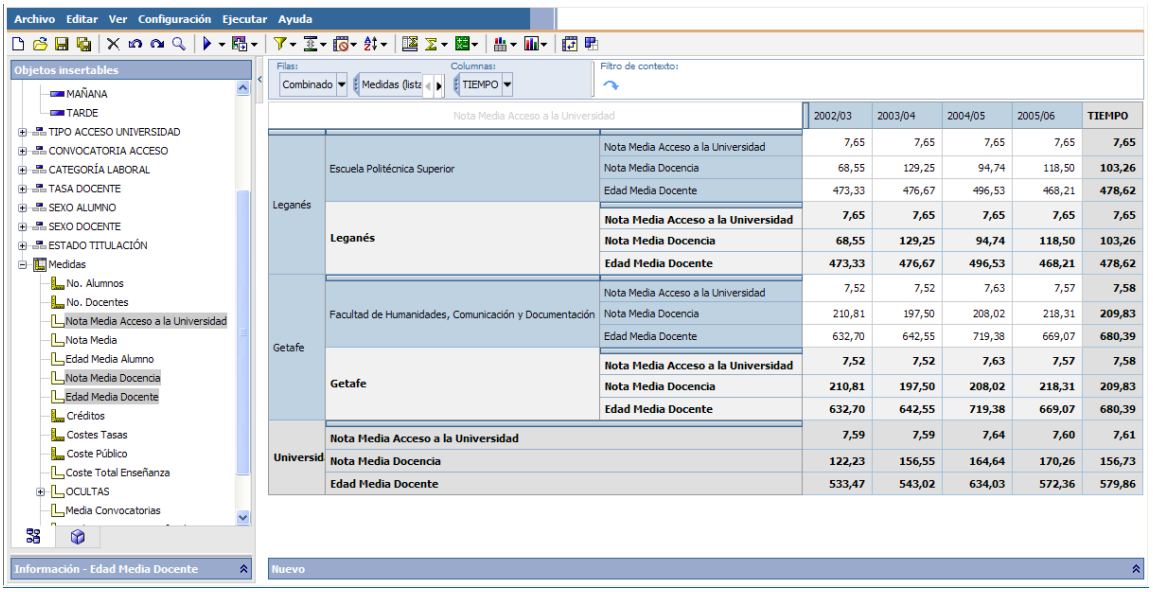


Ilustración 77. Resultado al realizar un nesting o anidamiento

Se pueden ocultar los subtotales, pulsando sobre ellos con el botón derecho y seleccionando la opción ocultar:

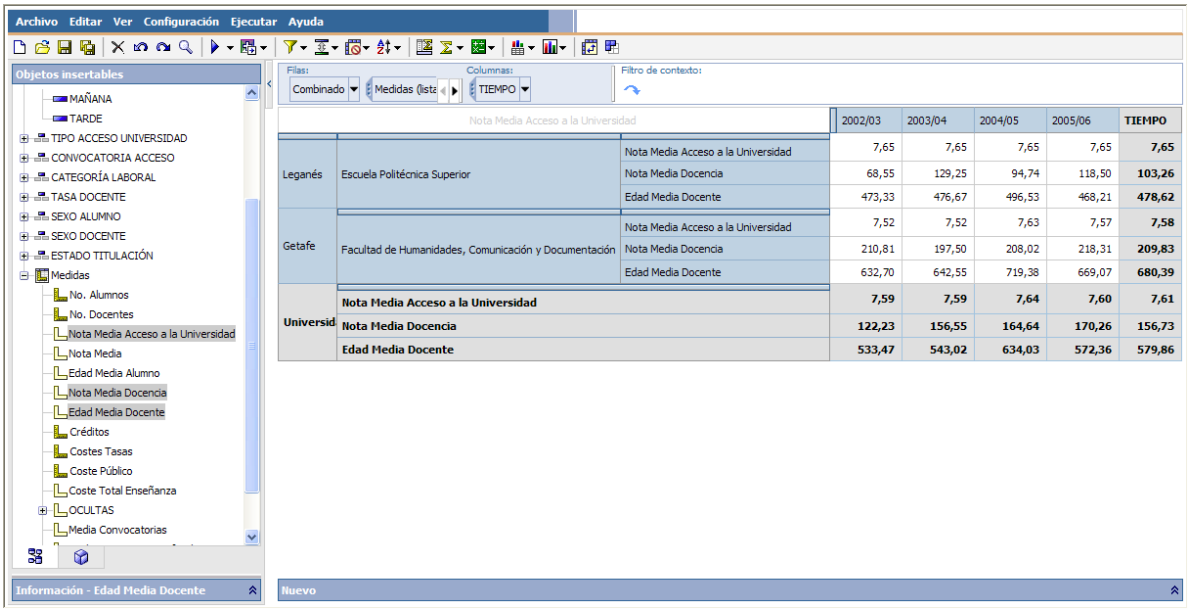


Ilustración 78. Ocultar subtotales en un nesting o anidamiento

A.5.8. Suprimir filas/columnas sin datos

Si mientras se realiza un análisis se encuentran filas y/o columnas sin datos (en blanco) la herramienta permite ocultarlas para facilitar la visualización.

Supongamos la obtención de estos resultados:

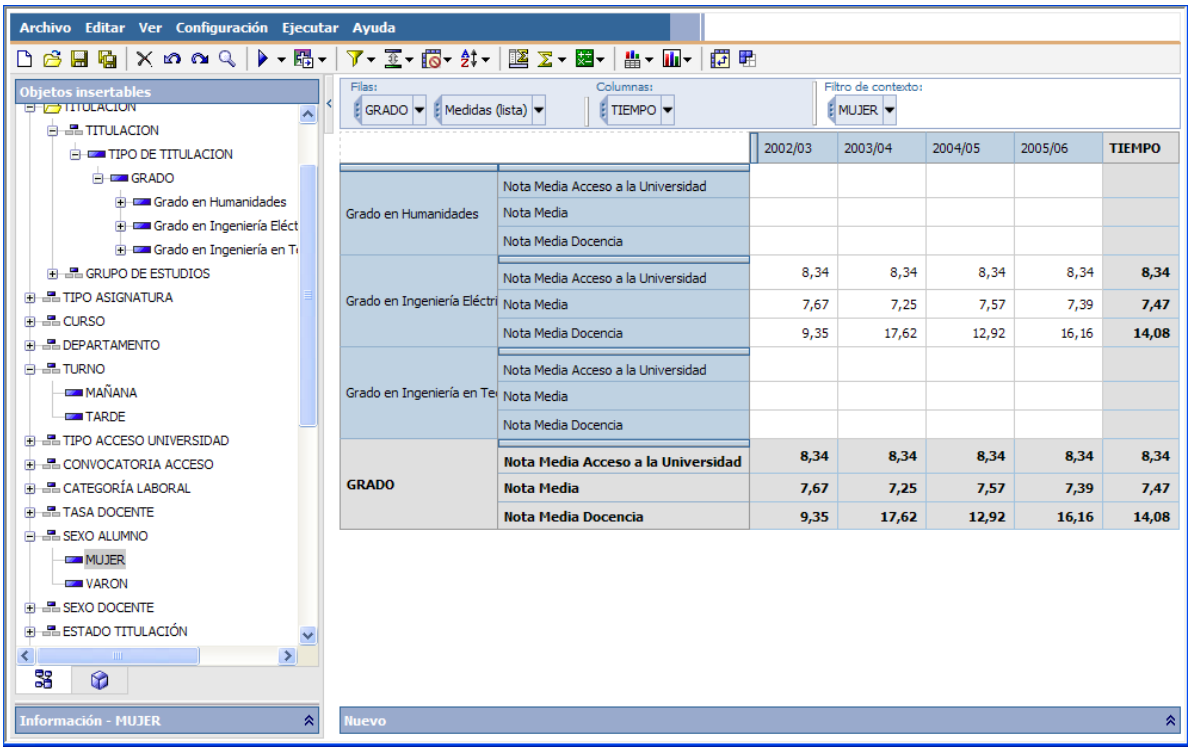


Ilustración 79. Suprimir filas sin datos

En este caso las filas referentes a los grados en Humanidades y Telecomunicaciones aparecen sin datos. Si queremos que las filas sin datos se oculten, pulsar el botón Suprimir  y seleccionar Suprimir sólo filas.

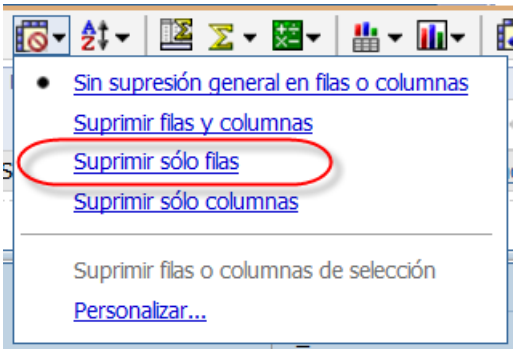


Ilustración 80. Suprimir filas

El resultado obtenido omite las filas sin datos. Mediante un aviso se indica que se está aplicando una supresión en filas. Pulsando sobre este aviso se eliminaría dicha supresión.

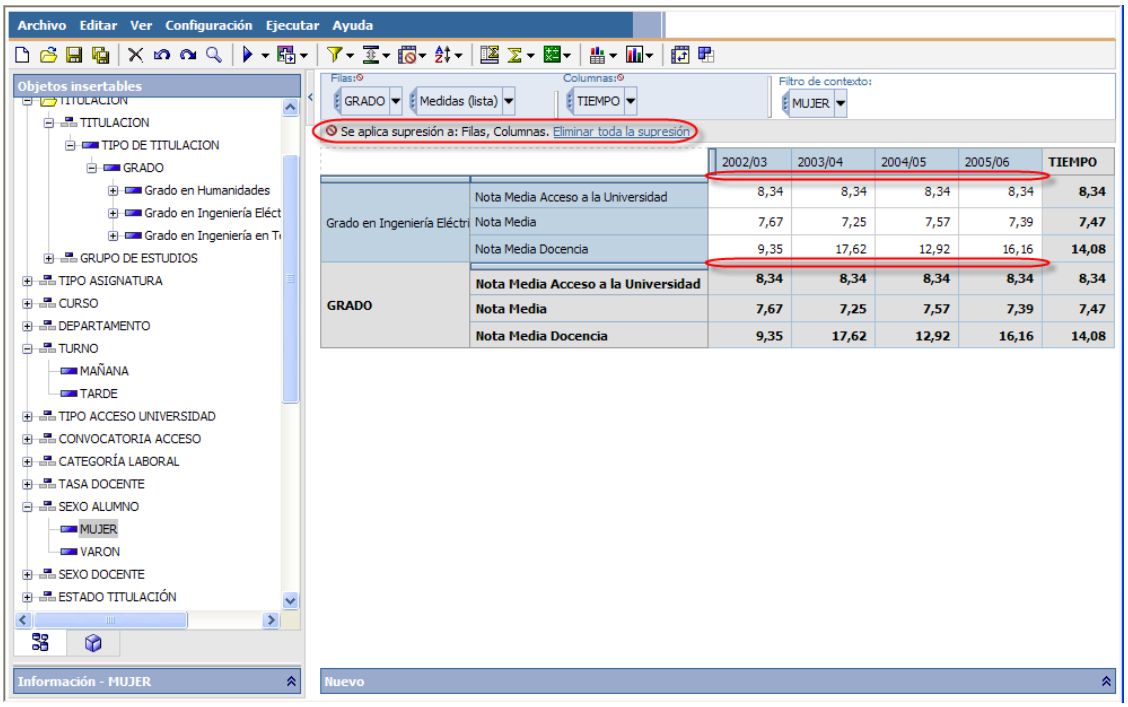


Ilustración 81. Resultado suprimir filas sin datos

A.5.9. Inserción de gráficos

En el cubo es posible complementar los análisis mediante gráficos. Se supone el siguiente análisis, al que se desea añadir un grafico de columnas.

Pulsar el botón Tipo de gráfico  . Seleccionar el tipo de gráfico deseado.

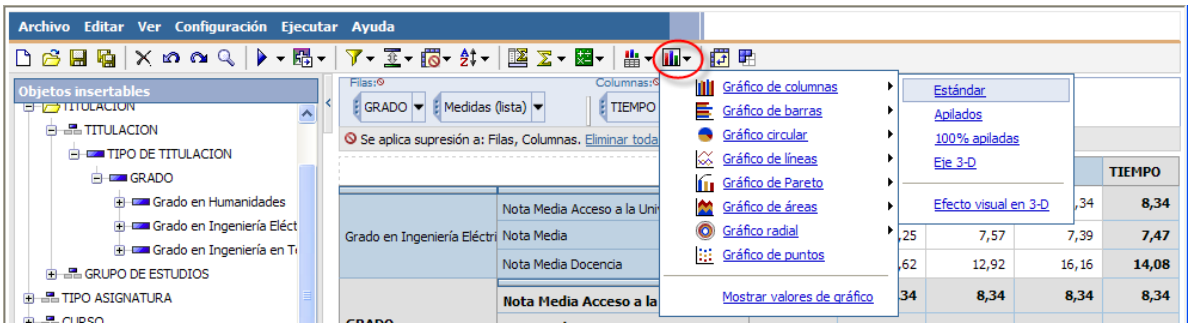


Ilustración 82. Insertar gráfico

Insertando un gráfico de columnas estándar se obtiene el siguiente resultado:

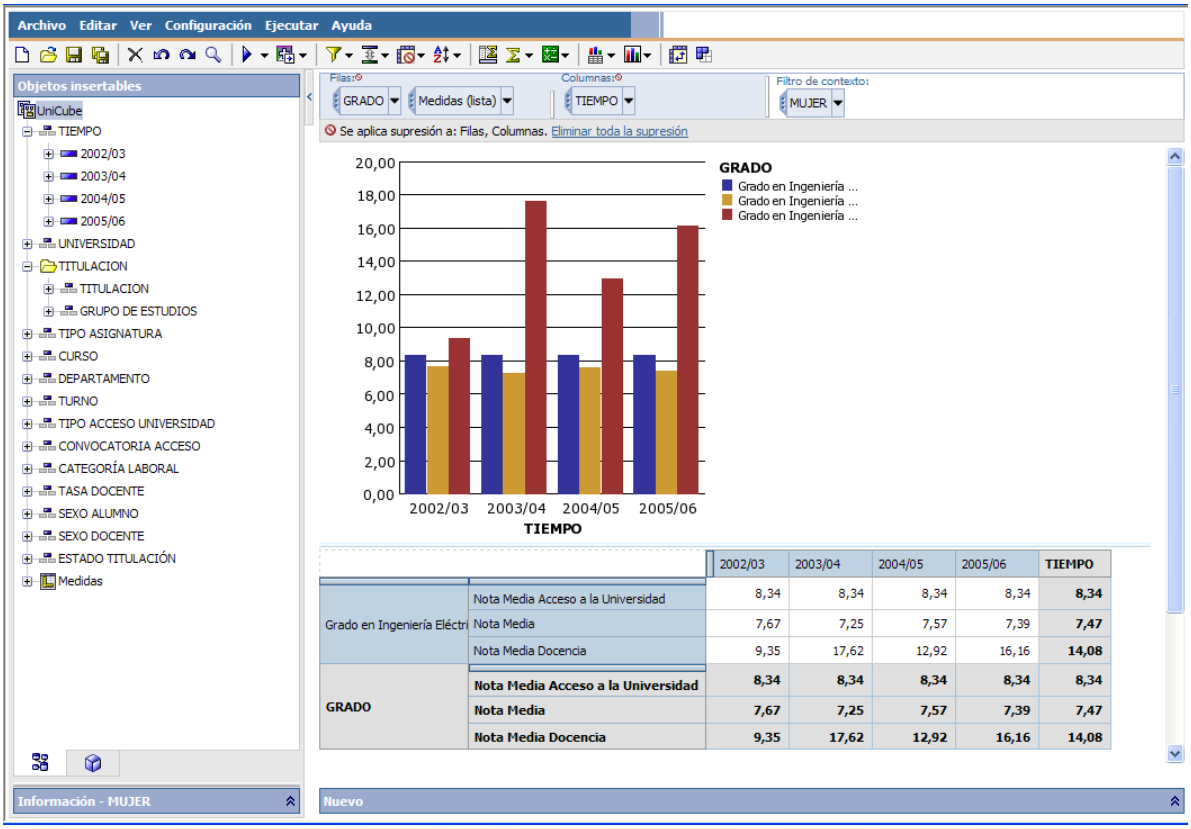


Ilustración 83. Resultado insertar gráfico

Se puede mostrar como resultado el gráfico con la tabla de valores cruzados, o bien el gráfico únicamente. Para ello pulsar el botón Mostrar y seleccionar Gráfico. El resultado será el gráfico, sin la tabla de datos en la parte inferior.

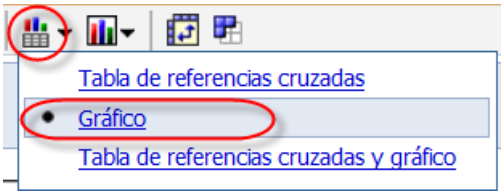


Ilustración 84. Mostrar sólo gráfico

A.5.10. Guardar un informe dinámico del cubo

Una vez que se completa el análisis deseado, es posible conservarlo para consultarlo periódicamente con los datos actualizados sin necesidad de construirlo desde cero.

Para guardar una vista del cubo, pulsar el botón Guardar como... en la barra de herramientas. Seleccionar la localización donde guardar el informe. Pulsar Guardar.

Se puede elegir entre dos localizaciones para guardar la vista:

- **Carpetas Públicas:** Las vistas que se guarden aquí son visibles por todos los usuarios. Por defecto no se cuenta con permisos para almacenar vistas en las carpetas públicas.

- **Mis Carpetas:** En esta carpeta personal se pueden guardar las vistas propias e incluso crear una estructura privada de directorios para organizar las distintas vistas.

En este ejemplo, se selecciona Mis Carpetas y se nombra "*Mi vista*".

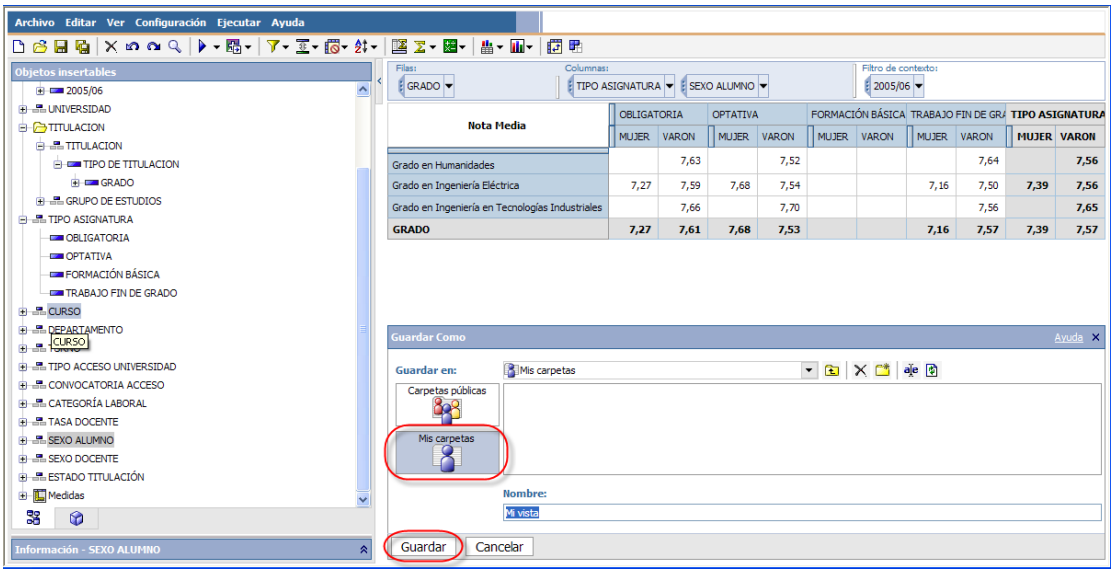


Ilustración 85. Guardar vista

Con esto nuestro informe quedará guardado en Mis Carpetas.

Para acceder a la vista de análisis existen dos caminos:

- Desde la ventana de análisis del cubo.

- Desde el Portal de Cognos.

Desde la ventana de análisis del cubo.

Pulsar el botón Abrir . Aparece en la parte inferior un menú donde buscar la vista deseada. Esta puede estar en Carpetas públicas o en Mis Carpetas. Una vez localizada, se selecciona y pulsa Abrir.

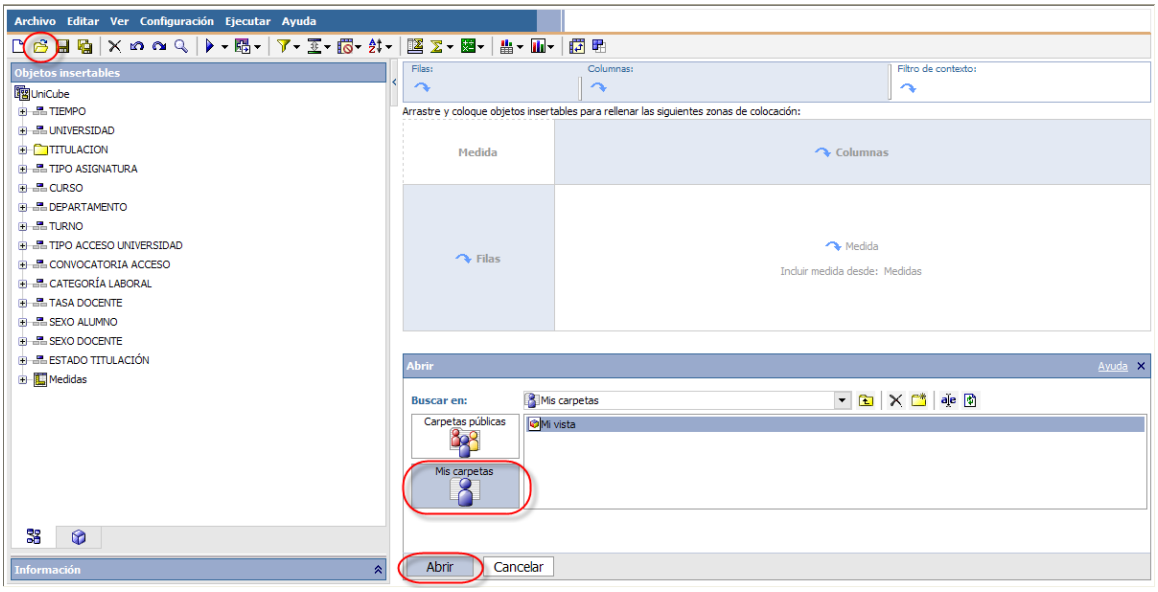


Ilustración 86. Abrir vista desde ventana de análisis

Desde el Portal de Cognos.

Acceder a Mi página principal.

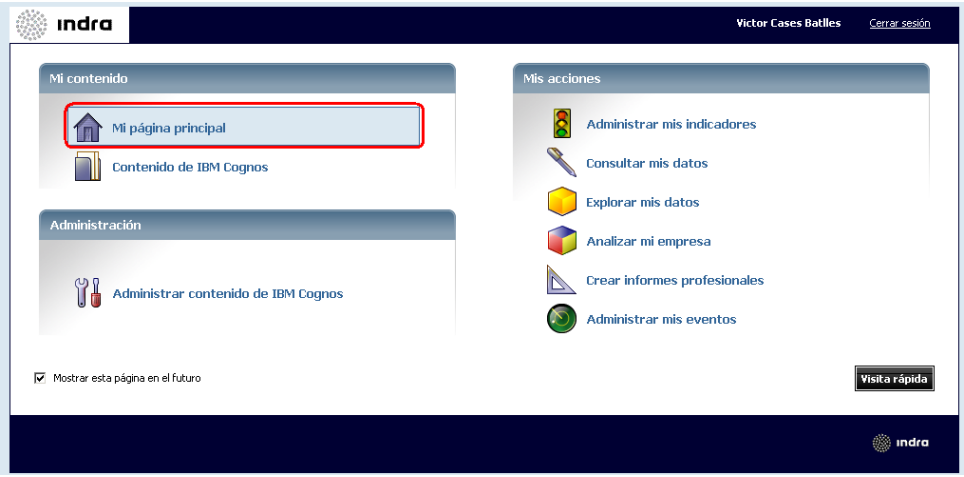


Ilustración 87. Página principal Cognos

Dentro de Mi página principal aparecen las pestañas Carpetas públicas y Mis Carpetas. Por defecto, se muestra Carpetas públicas. Acceder a Mis Carpetas. Pinchando sobre el informe guardado se abre con los datos actualizados.

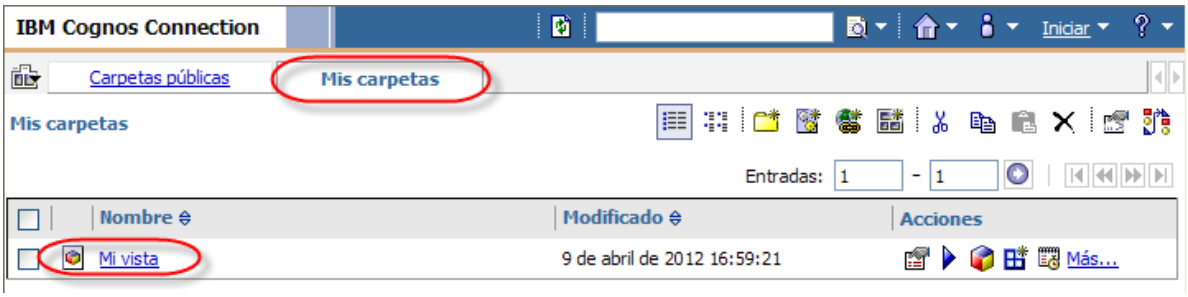


Ilustración 88. Abrir vista desde Mis carpetas

A.6. Casos prácticos

En este capítulo se realizará una simulación de un análisis para poner en práctica lo descrito en los tres capítulos anteriores.

En este ejemplo, se utilizarán las acciones principales para, poco a poco, ir añadiendo funciones más avanzadas. Además, se tratarán algunos errores comunes en la construcción de informes.

A.6.1. Ejemplo de análisis básico

Este análisis comenzará con un análisis simple con los siguientes objetivos:

- Obtener la nota media.
- Sólo para el curso académico 2010/11.
- Sólo para las titulaciones de grado.
- Ver esta información por curso.

Se realizará de la siguiente manera:

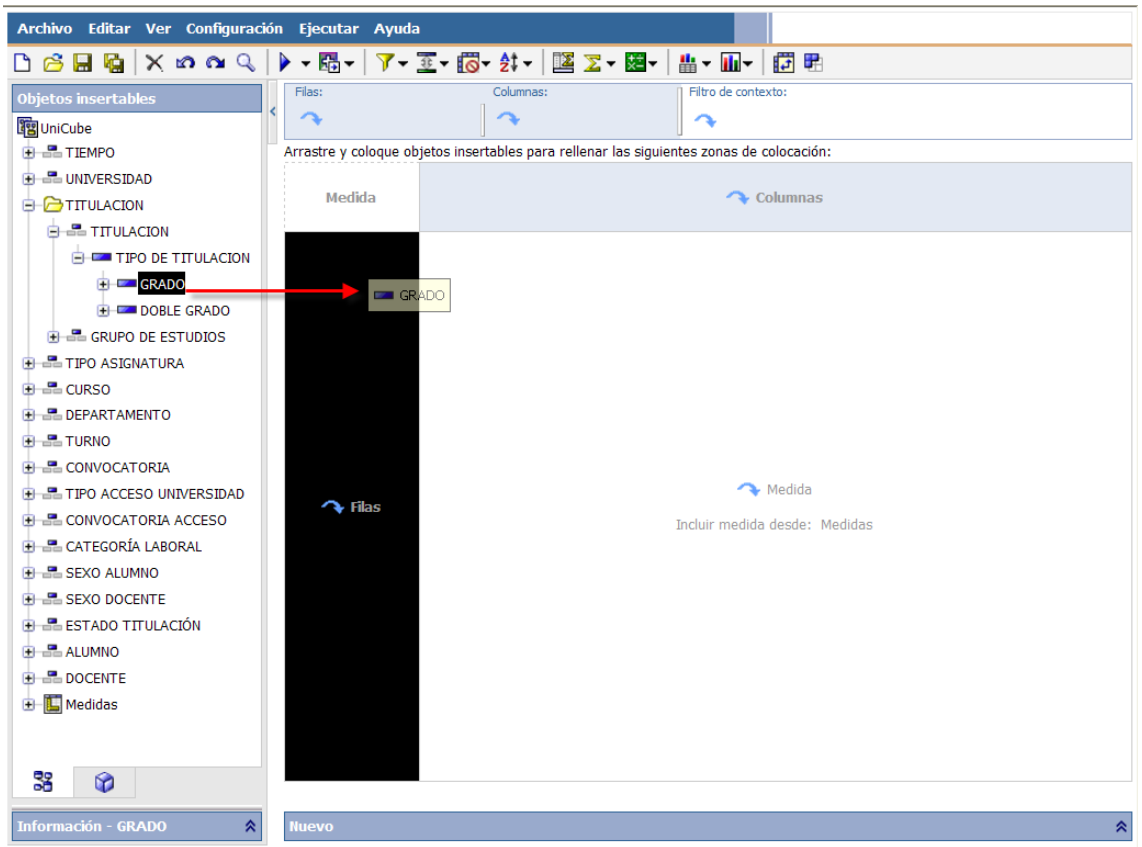
1. Colocar Grado en las filas.
2. Colocar el Curso en las columnas.

3. Introducir la medida “Nota media”.

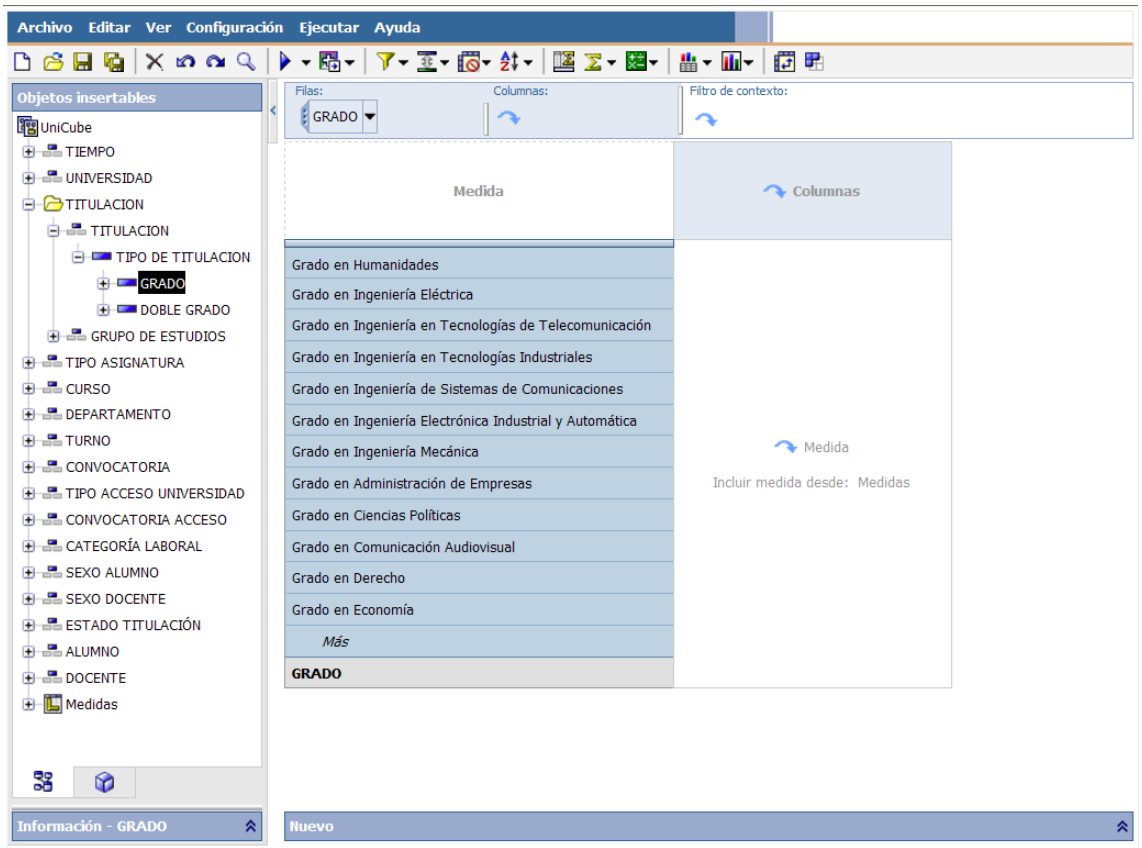
4. Filtrar los datos para el año 2010/11.

1. Colocar Grado en las filas

Las titulaciones de Grado son un tipo de titulación, por lo tanto, se despliega en la dimensión de Titulación por el camino Tipo de Titulación y arrastrar Grado al área donde se emplazan las filas.

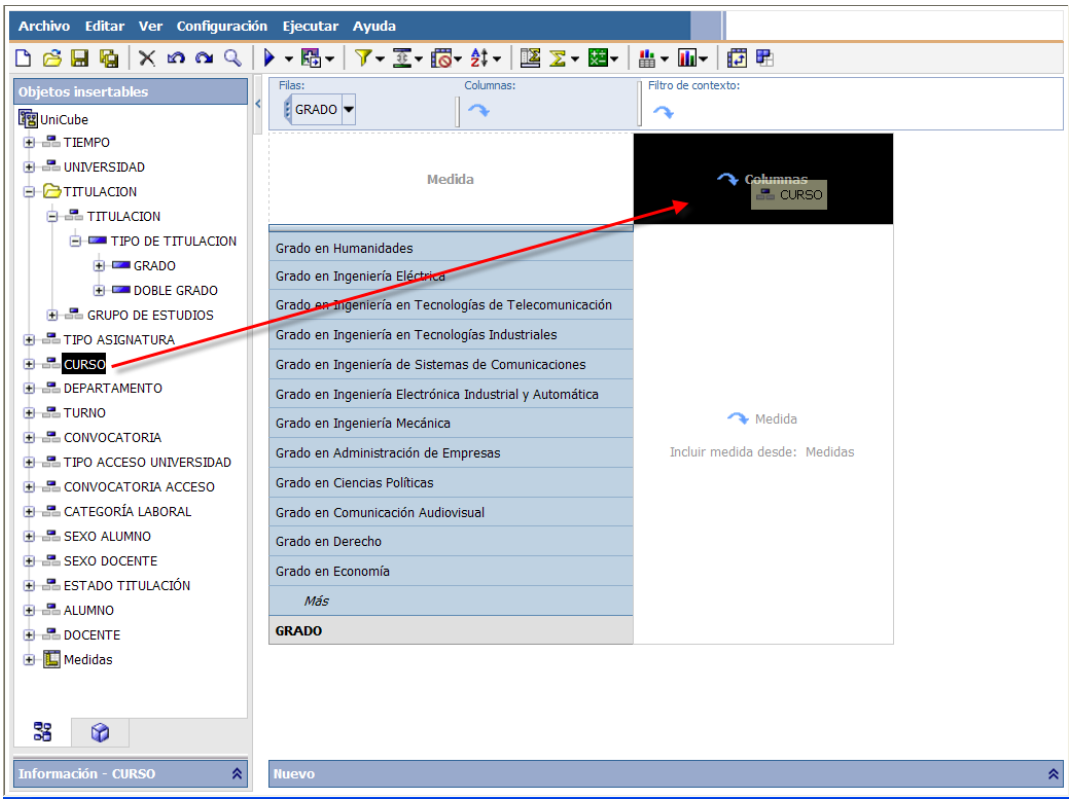


Con esto, todas las titulaciones de este tipo figuran en las filas:

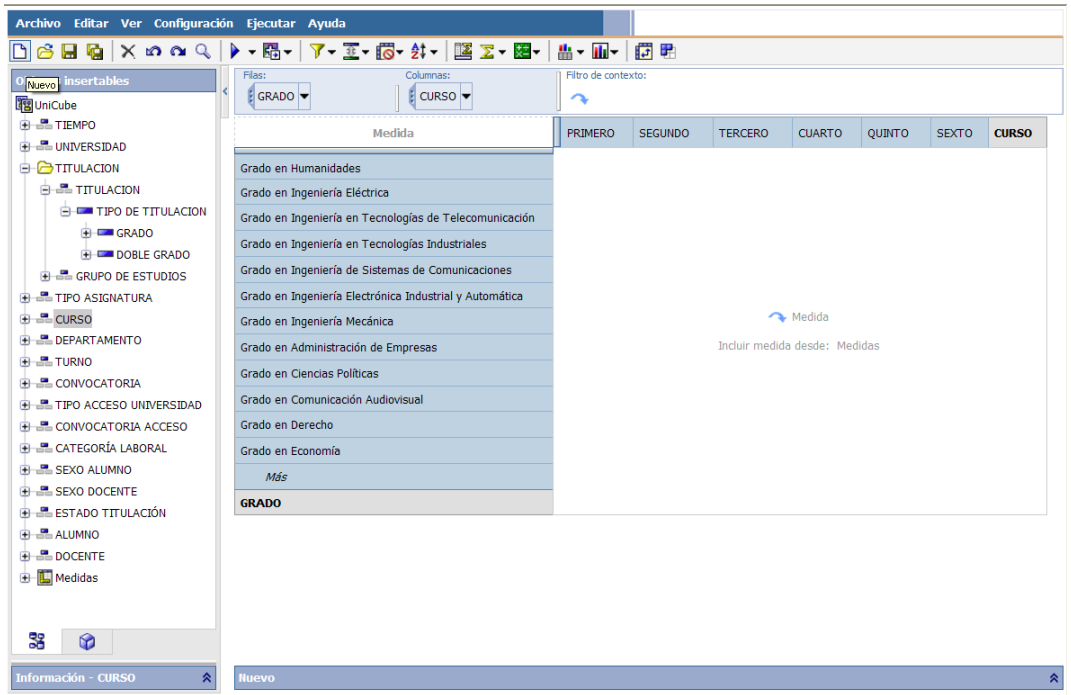


2. Colocar el Curso en las columnas

Desplegar la Dimensión de Curso, donde se observan los distintos cursos. Como no se desea arrastrar ninguno en particular sino todos, arrastrar la dimensión.



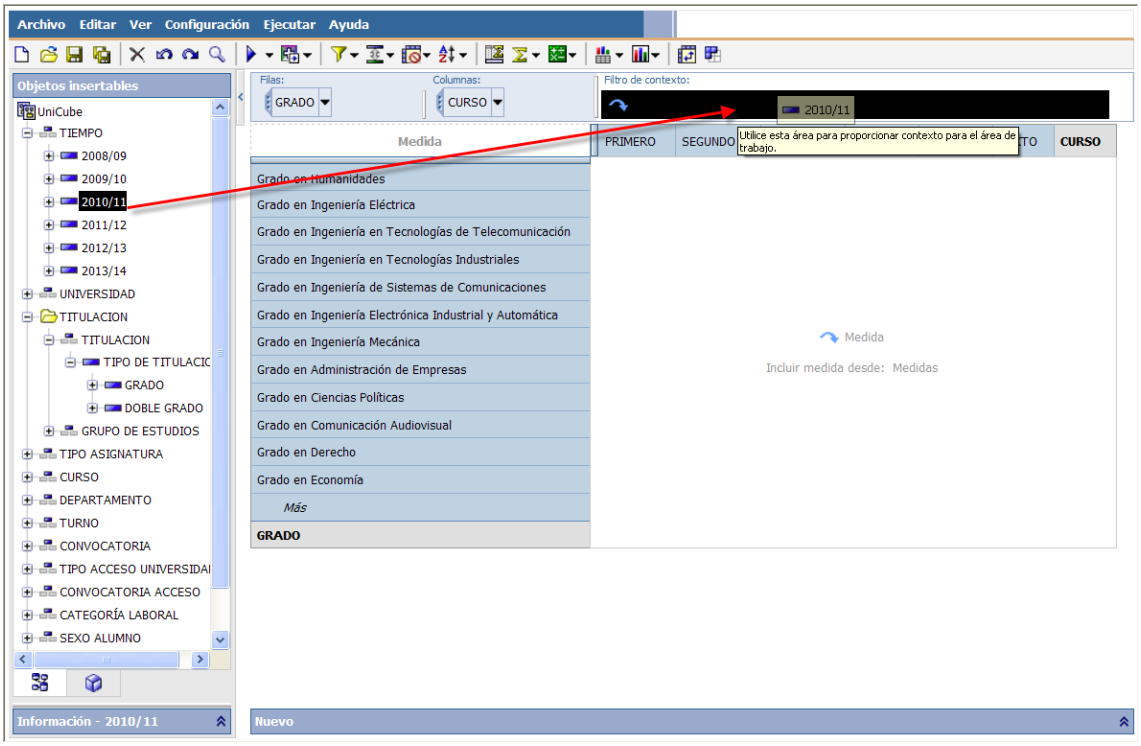
Se obtiene el siguiente resultado:



En el panel superior se indica que se está mostrando en filas Grado y en las columnas el Curso. Este panel es muy útil para conocer en todo momento qué datos estamos visualizando, sobre todo cuando el estudio va adquiriendo complejidad.

3. Filtrar los datos para el año académico 2010/11

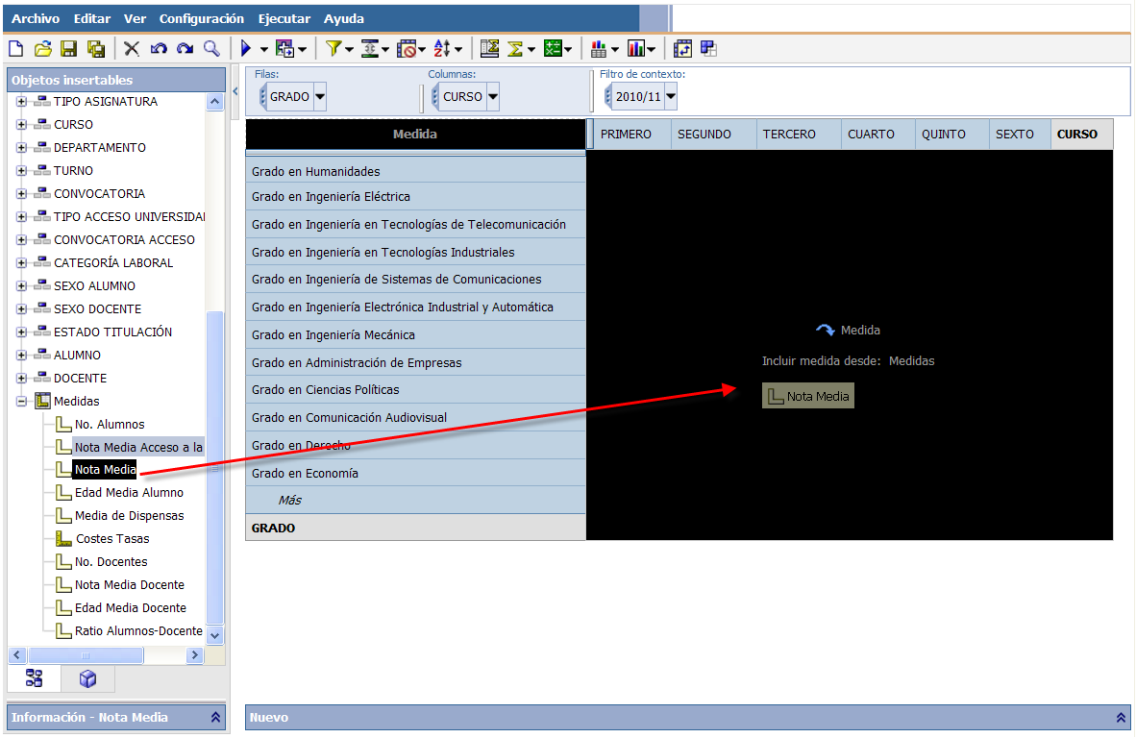
Seleccionar 2010/11 en la dimensión tiempo y arrastrar hasta el área de filtro.



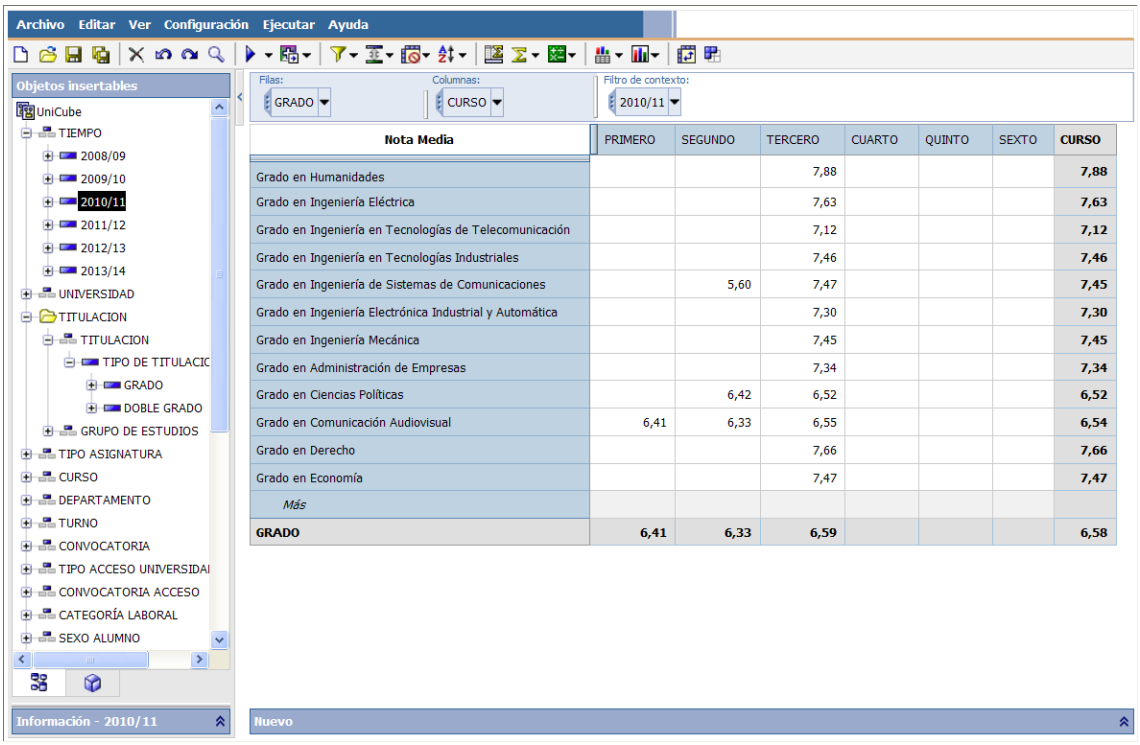
De esta manera, tanto la estructura de la tabla como la medida son las mismas pero los datos mostrados corresponden a la media de los cuatrimestres de 2010/11. Antes, sin filtro de fecha, se visualizaba toda la historia registrada de Grado.

4. Introducir la medida "Nota media"

Para introducir esta medida, desplegar las Medidas, localizar Nota media y arrastrar hacia el área central de la tabla, donde se muestran los datos.



Obteniendo los datos para esta medida:



A.6.2. Ejemplo de análisis medio

En este caso práctico se realizará un análisis un poco más avanzado, con los siguientes objetivos:

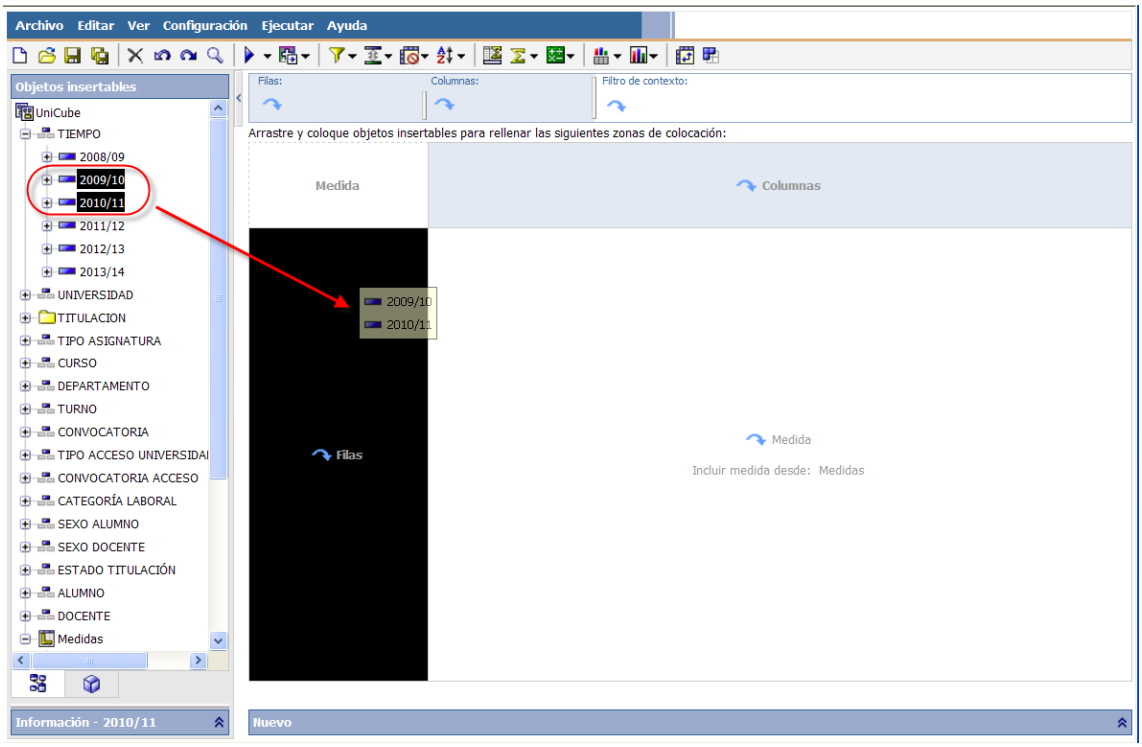
- Obtener el número de alumnos de todas las titulaciones de grado de la Universidad Carlos III de Madrid.
- Obteniendo todos los datos de 2009/10 y 2010/11, desglosado por cuatrimestres.
- Sólo para el campus de Leganés.
- Desglosado por centros.
- Obtener un PDF para enviar por correo el informe obtenido.

Los pasos a seguir son los siguientes:

1. Colocar los años académicos 2009/10 y 2010/11 en las columnas.
2. Expandir el tiempo por cuatrimestres.
3. Colocar el nivel centros del campus "Leganés".
4. Anidar las titulaciones de grado en las filas.
5. Introducir la medida Número de alumnos.
6. Exportar a PDF.

1. <u>Colocar los años académicos 2009/10 y 2010/11 en las columnas.</u>

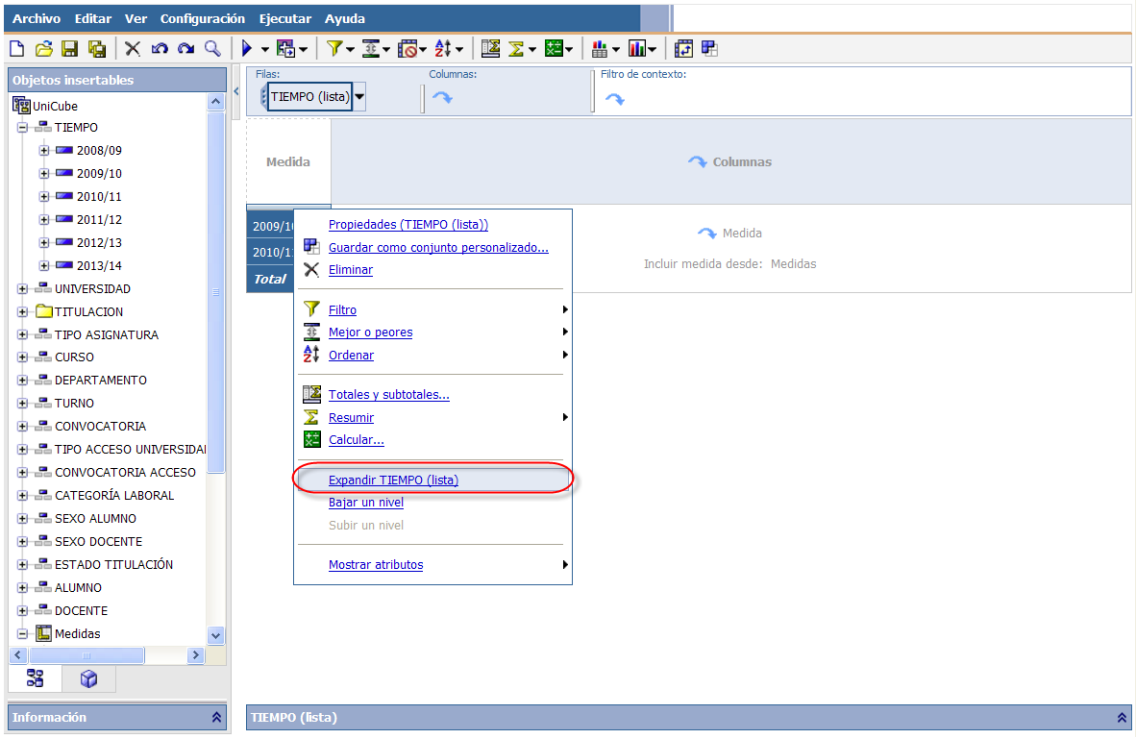
Localizar los años académicos 2009/10 y 2010/11 en la dimensión de tiempo, e insertar como filas.



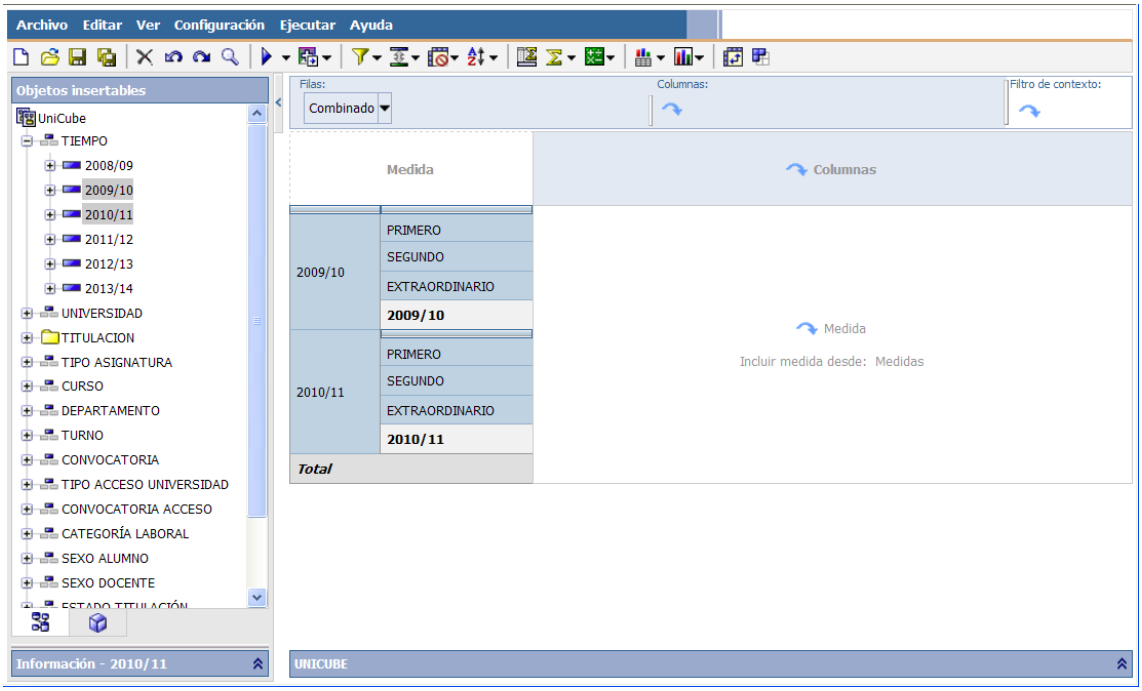
En muchos casos, para seleccionar varios elementos tenemos que mantener pulsada la tecla Ctrl.

2. Expandir el tiempo por cuatrimestres.

Aparecen los años académicos, para bajar al nivel de cuatrimestre sin perder la referencia al año, expandir Tiempo.

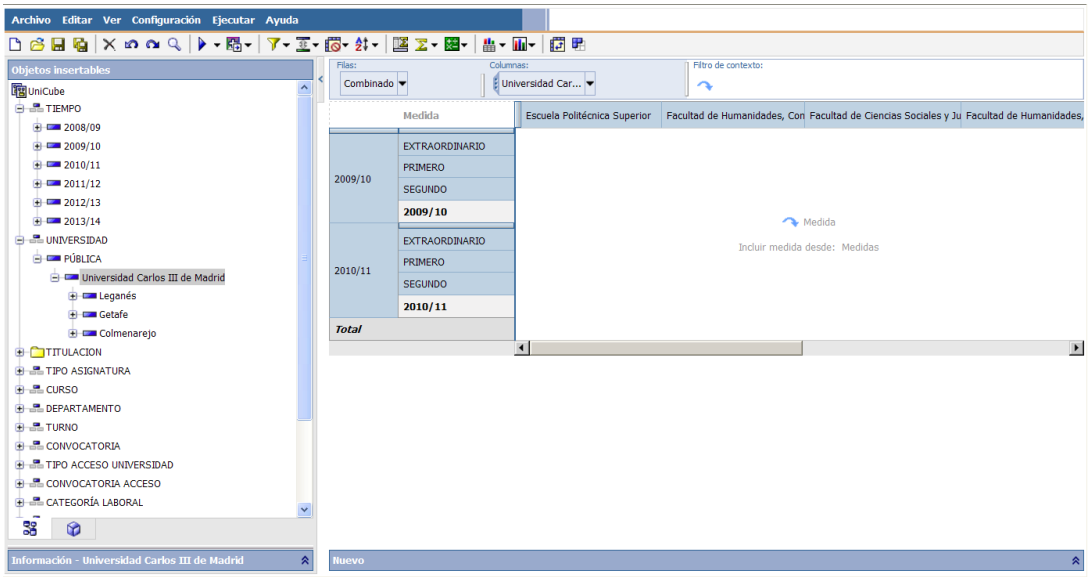


De los pasos 1 y 2 se obtiene el siguiente resultado:



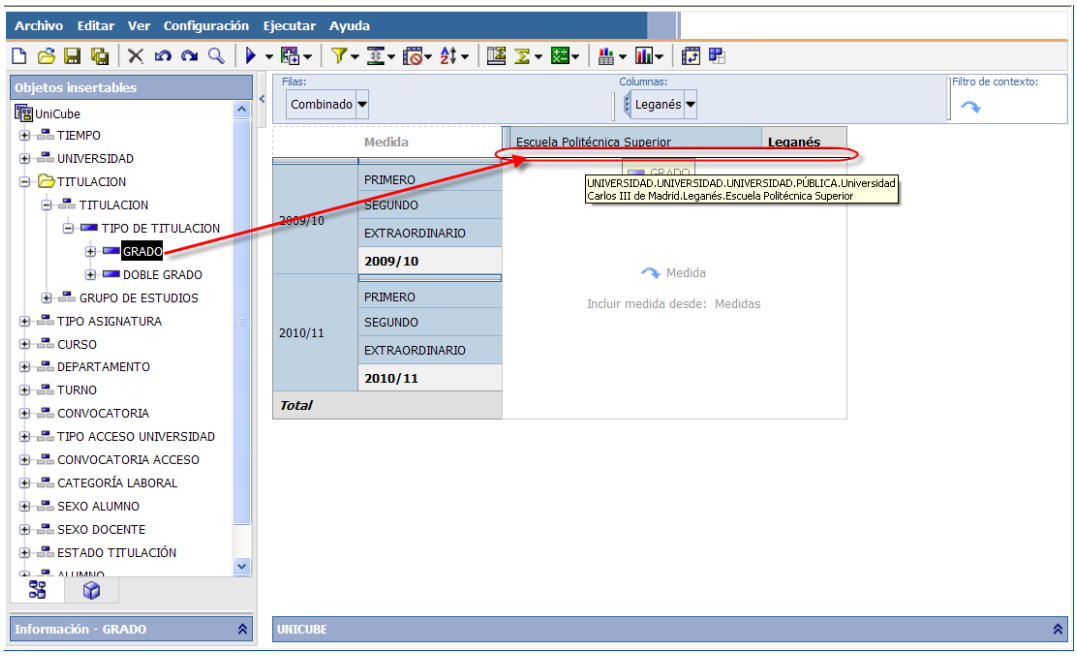
3. Colocar el nivel centros del campus "Leganés".

Arrastrar la Universidad Carlos III de Madrid al área de anidamiento de las filas, aparecerá el nivel inmediatamente inferior, los campus. Como se debe llegar al nivel de centros, pulsar Bajar un nivel. Se obtiene así el desglose deseado:



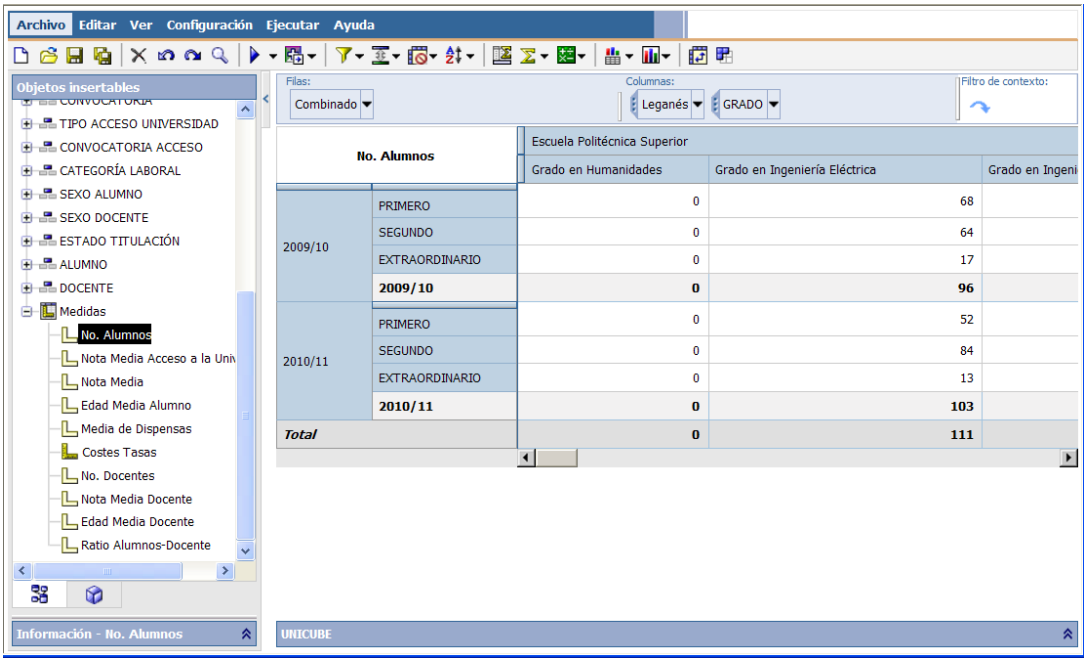
4. Anidar las titulaciones de grado en las filas.

Arrastrar al área de anidamiento el tipo de titulación grado, desplegándose así el nivel inferior, titulaciones, y consiguiendo de esta forma es desglose especificado.

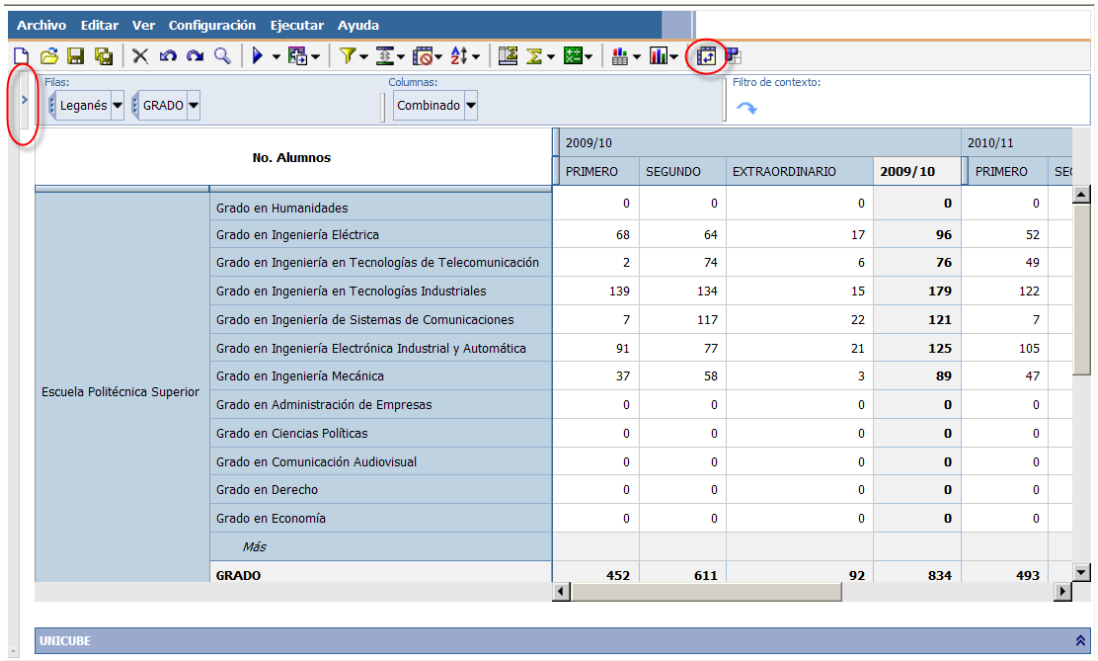


5. Introducir la medida Número de alumnos.

Seleccionar la medida No. Alumnos de la carpeta de medidas y arrastrar al área central de la tabla, obteniendo los datos:



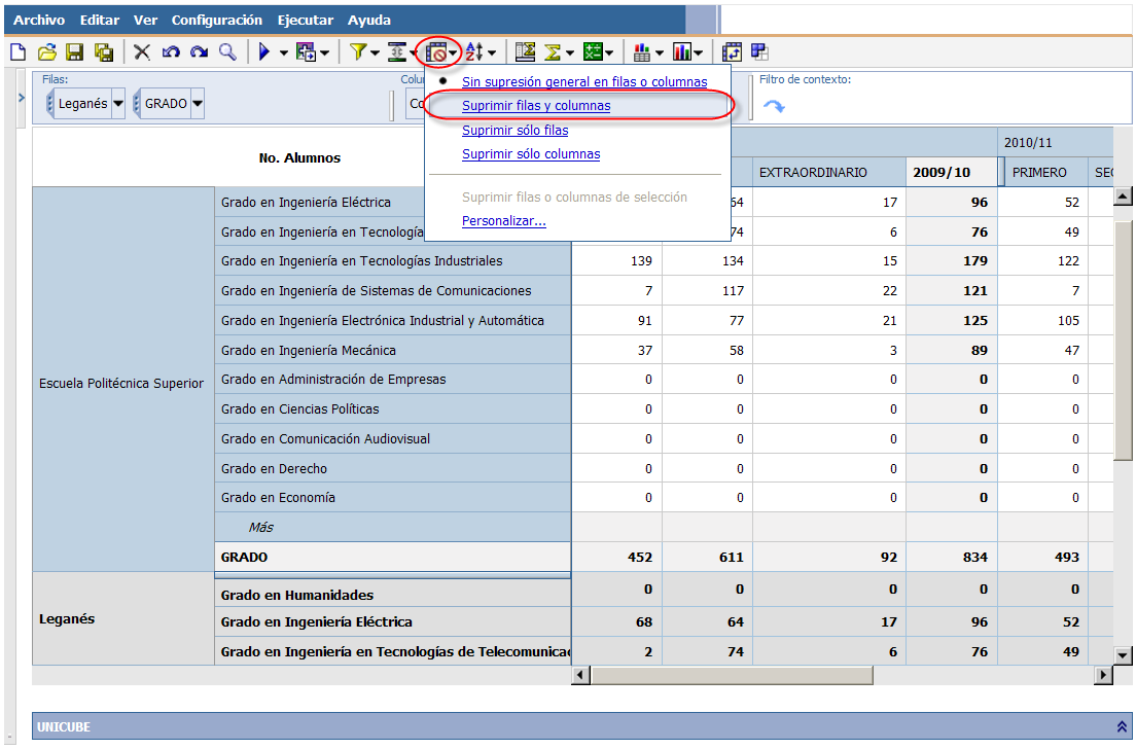
Se observa que es necesario el desplazamiento lateral continuamente, es posible ganar espacio ocultando el panel de objetos insertables, pero lo mejor es pivotar la tabla, utilizando el botón asociado a esta operación:



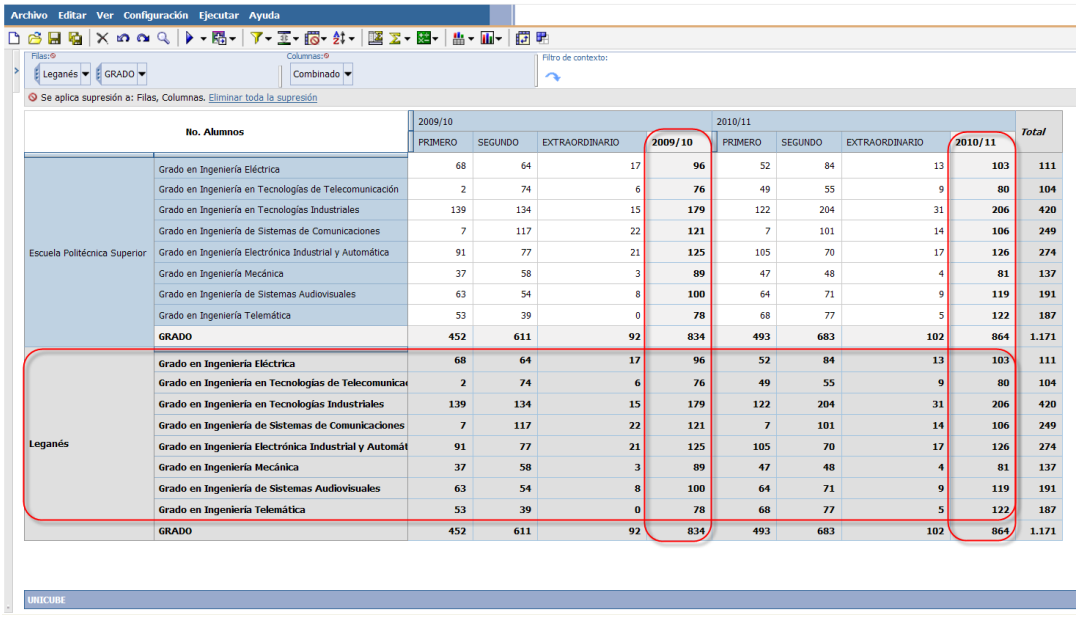
Se observan muchas filas con valores 0, esto ocurre porque al anidar los grados se han seleccionado todos ellos, independientemente o no de haber seleccionado los centros de 'Leganes'.

> En la última fila mostrada, antes del Total GRADO, se muestra un "Más".
>
> Esto indica que no se están visualizando todas las titulaciones de Grado. Pulsando en "Más" se puede seleccionar el número de filas que se muestran en el análisis. Seleccionando máximo, aparecen todas.

Eliminando las filas y columna sin valores se eliminan aquellos para los que no hay alumnos. Teniendo los datos de sólo 'Leganés'.



Los anidamientos crean automáticamente un subtotal, en este análisis se oculta pulsando sobre un subtotal con el botón derecho del ratón y seleccionando "Ocultar".



No. Alumnos		2009/10				2010/11				Total
		PRIMERO	SEGUNDO	EXTRAORDINARIO	2009/10	PRIMERO	SEGUNDO	EXTRAORDINARIO	2010/11	
Escuela Politécnica Superior	Grado en Ingeniería Eléctrica	68	64	17	96	52	84	13	103	111
	Grado en Ingeniería en Tecnologías de Telecomunicación	2	74	6	76	49	55	9	80	104
	Grado en Ingeniería en Tecnologías Industriales	139	134	15	179	122	204	31	206	420
	Grado en Ingeniería de Sistemas de Comunicaciones	7	117	22	121	7	101	14	106	249
	Grado en Ingeniería Electrónica Industrial y Automática	91	77	21	125	105	70	17	126	274
	Grado en Ingeniería Mecánica	37	58	3	89	47	48	4	81	137
	Grado en Ingeniería de Sistemas Audiovisuales	63	54	8	100	64	71	9	119	191
	Grado en Ingeniería Telemática	53	39	0	78	68	77	5	122	187
	GRADO	452	611	92	834	493	683	102	864	1.171
Leganés	Grado en Ingeniería Eléctrica	68	64	17	96	52	84	13	103	111
	Grado en Ingeniería en Tecnologías de Telecomunica	2	74	6	76	49	55	9	80	104
	Grado en Ingeniería en Tecnologías Industriales	139	134	15	179	122	204	31	206	420
	Grado en Ingeniería de Sistemas de Comunicaciones	7	117	22	121	7	101	14	106	249
	Grado en Ingeniería Electrónica Industrial y Automát	91	77	21	125	105	70	17	126	274
	Grado en Ingeniería Mecánica	37	58	3	89	47	48	4	81	137
	Grado en Ingeniería de Sistemas Audiovisuales	63	54	8	100	64	71	9	119	191
	Grado en Ingeniería Telemática	53	39	0	78	68	77	5	122	187
	GRADO	452	611	92	834	493	683	102	864	1.171

En mitad de un análisis como este, en el que el volumen de datos es considerable y hacemos cambios en la estructura de la tabla, como ocultar un subtotal. La operación puede tardar más de lo esperado.
Lo más óptimo para realizar un análisis ágil es definir bien la estructura (filas, columnas y filtros) y en un último paso añadir la/las medida/s.

6. Exportar a PDF.

Una vez la vista es la deseada, se exporta a PDF. Pulsando en el botón "Ejecutar Informe" y seleccionando "Ejecutar Informe (PDF)".

Una vez abierto el informe como PDF, es posible guardarlo en el directorio deseado.

A.6.3. Ejemplos de análisis avanzado

El ejemplo a realizar constituye un análisis breve que incluye funciones que no suelen ser de uso cotidiano pero que pueden resultar útiles en algunos casos.

- Obtener las 10 mejores notas de la docencia.
- Sólo para datos de 2010/11.
- Sólo para la categoría de departamento 'Derecho'.
- Extraer un gráfico de barras con los datos.

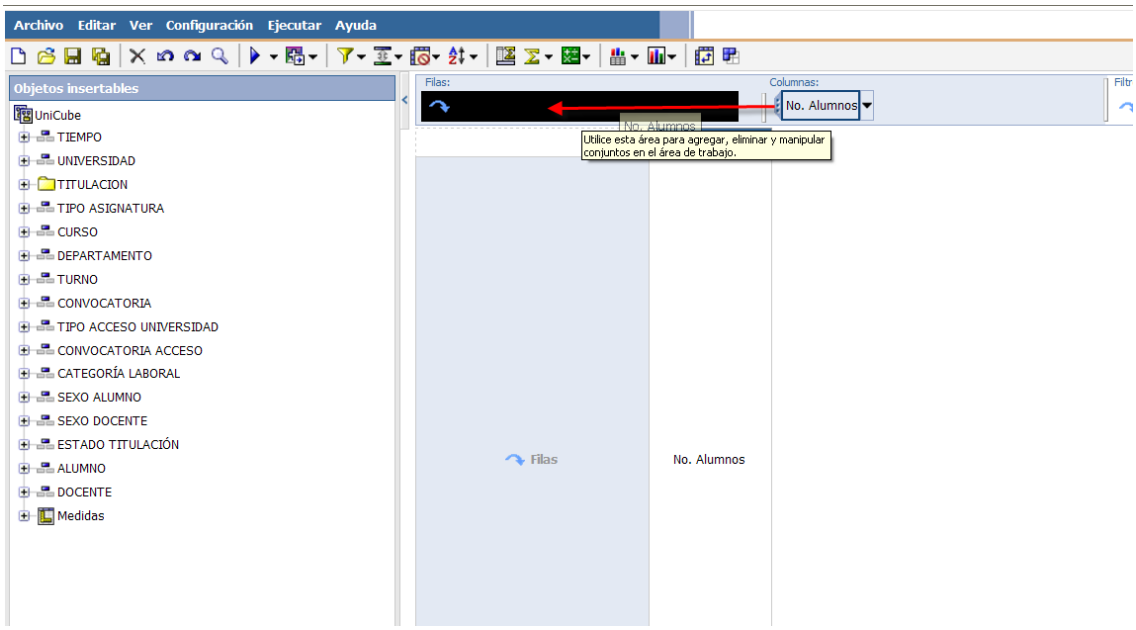
Se realizará de la siguiente manera:

1. Colocación de la medida Nota media Docencia en las columnas.
2. Colocación de la dimensión Docente en las filas.
3. Filtro para obtener datos de 2010/11.
4. Filtro para obtener datos de la categoría de departamento 'Derecho'
5. Cálculo del rango de los 10 mejores.
6. Creación de los gráficos para ver los datos.

1. <u>Colocación de la medida Nota media Docencia en las columnas.</u>

Arrastrar la medida al espacio de columnas.

Es posible cometer un error al arrastrar las dimensiones, para deshacer el cambio existe la opción en la barra de herramientas, pero es posible que el cambio de una dimensión de las filas a las columna se realice a mitad de un análisis, con lo que deshacer no es una buena opción. Las dimensiones y filtros pueden arrastrarse cómodamente en el panel superior.



2. **Colocación de la dimensión Docente en las filas.**

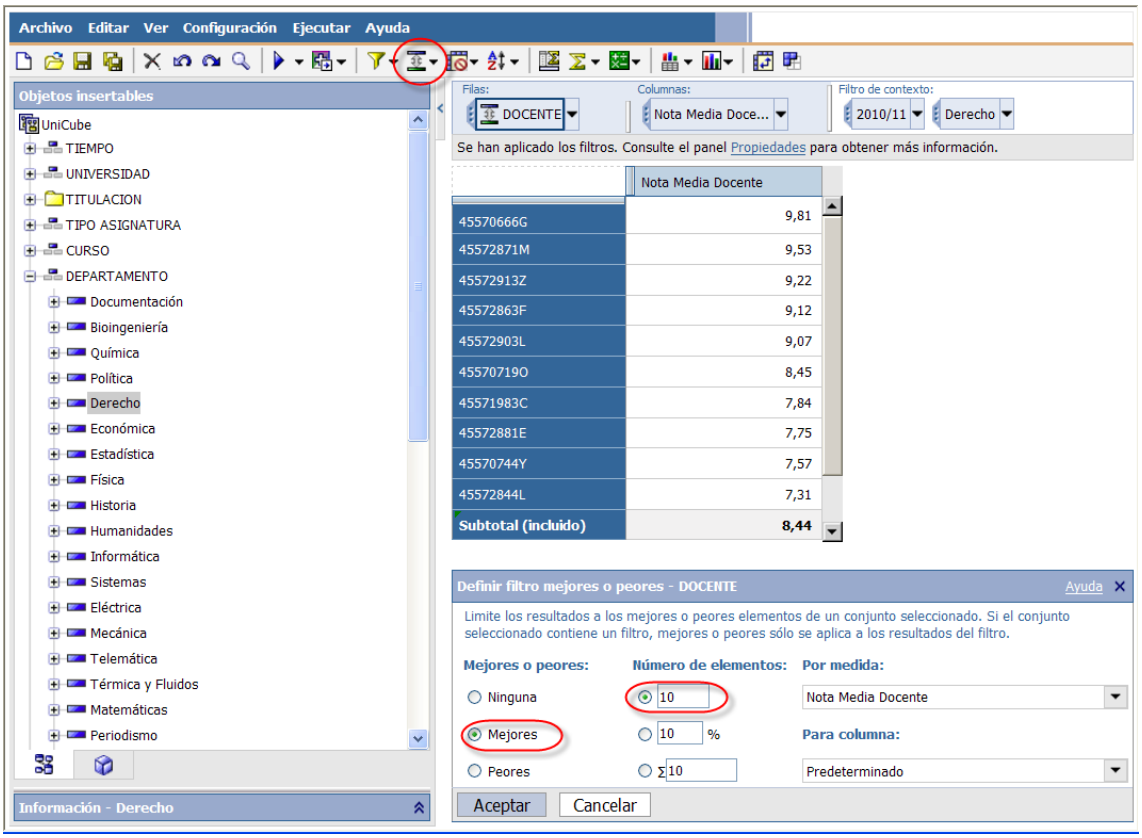
3. **Filtro para obtener datos de 2010/11.**

4. **Filtro para obtener datos de la categoría de departamento 'Derecho'**

5. **Cálculo del rango de los 10 mejores.**

6. **Creación de los gráficos para ver los datos.**

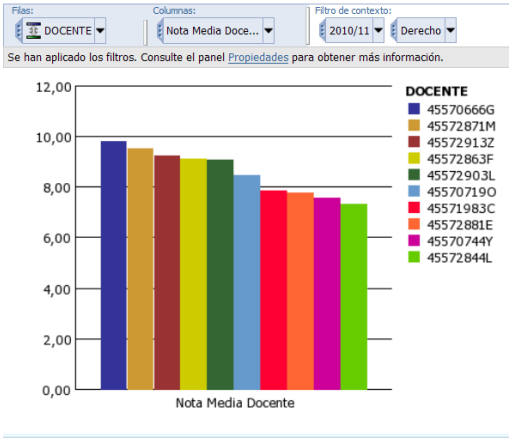
Los pasos del 2 al 4 son análogos a los realizados en los ejemplos anteriores. Para el cálculo del rango pulsar sobre la columna y elegir la opción de la barra de herramientas Mejores y peores > Personalizar. Detallar el número de elementos y la opción mejores.



El resultado muestra las mejores 10 notas.

7. Creación de los gráficos para ver los datos.

Insertar un gráfico de barras que muestre los datos. La opción esta presente en la barra de herramientas. Visualmente se observa la diferencia entre el mejor y el peor de los 10 docentes. Siendo la diferencia de 1,5 puntos aproximadamente.

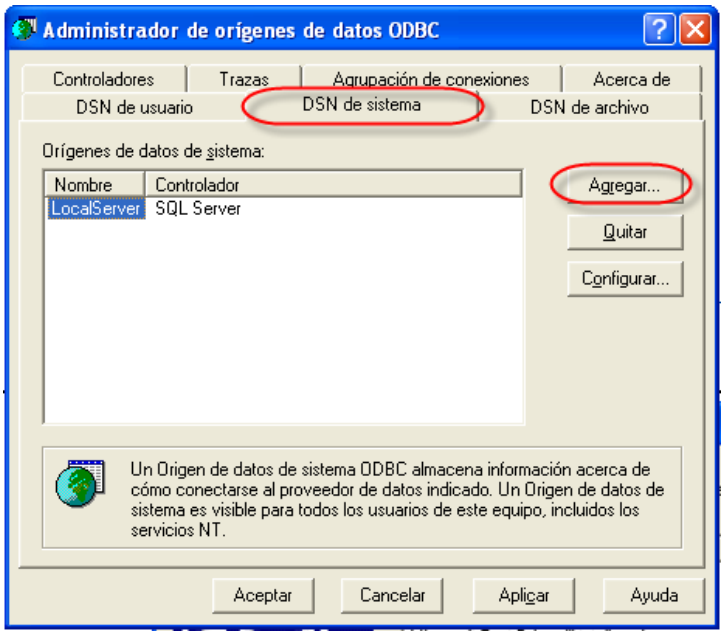


ANEXO B - Creación y configuración de un origen de datos ODBC

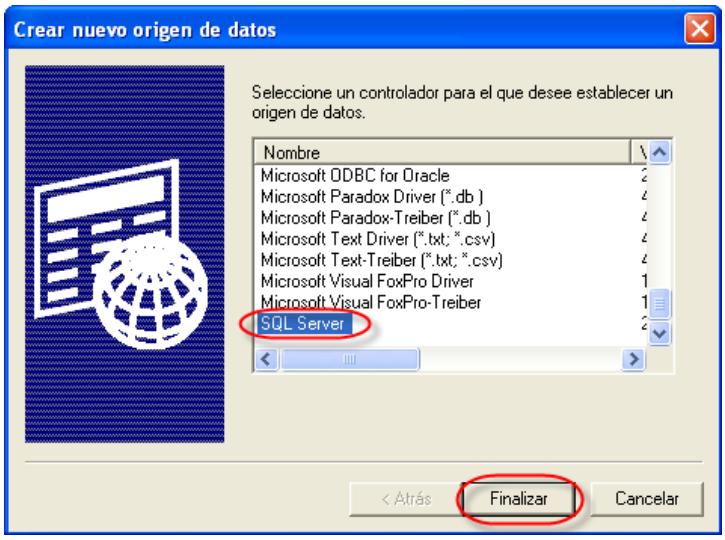
Esta configuración está basada en Windwos XP y un motor de BD SQL Server 2000.

Se accede a Orígenes de Datos (ODBC) en Herramientas administrativas del panel de control: Inicio / Configuración / Panel de control / Herramientas administrativas / Orígenes de Datos (ODBC)

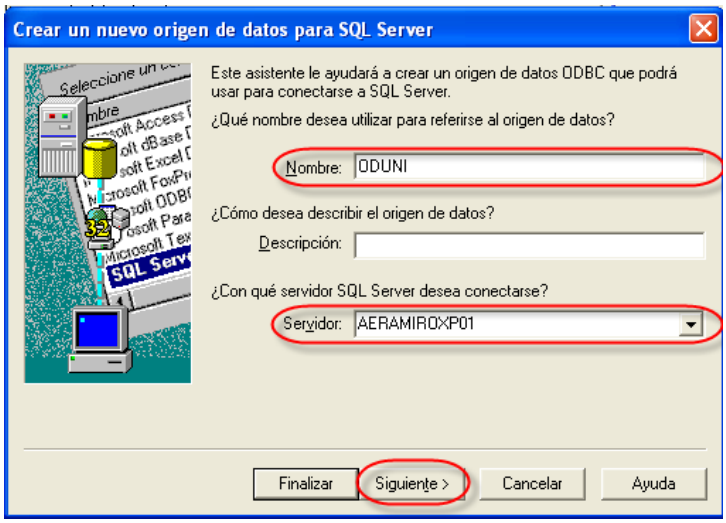
En DNS de sistema se agrega un nuevo origen de datos: Pestaña DNS de sistema. Botón Agregar…



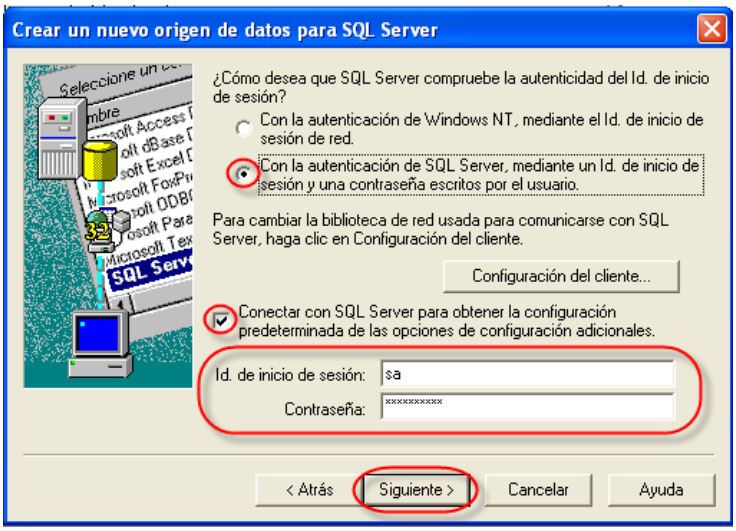
Se selecciona el controlador SQL Server y pulsar Finalizar.



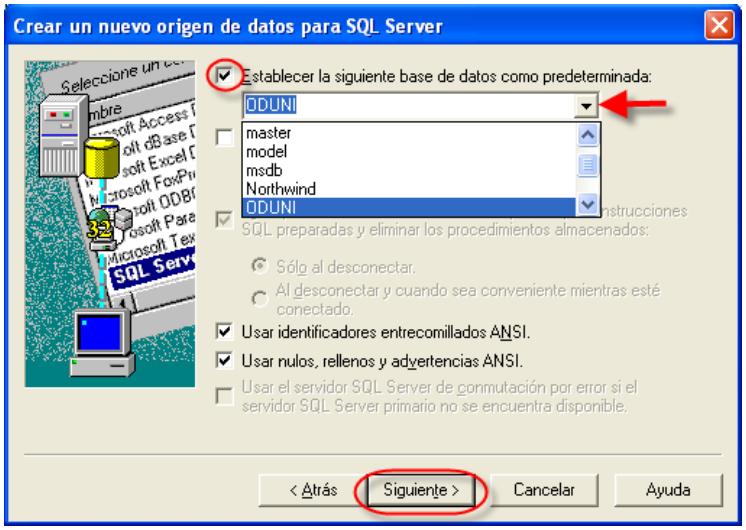
En la siguiente pantalla se rellena el nombre de la base de datos, el servidor y una descripción:



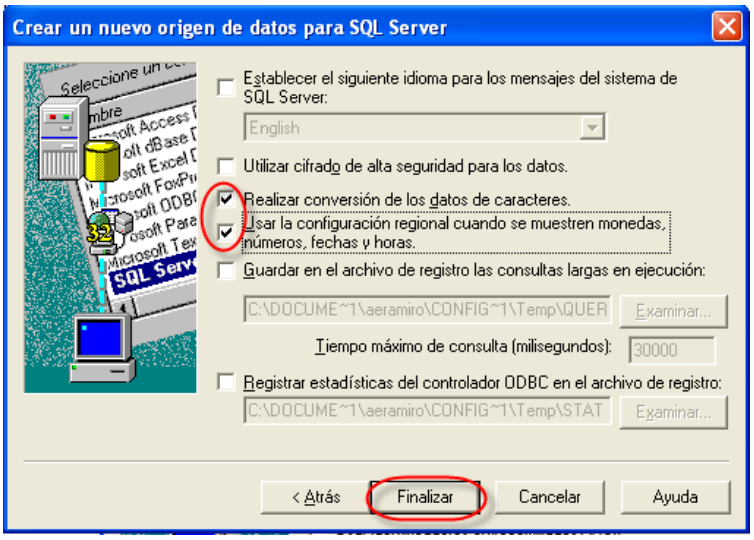
En el caso de tener una autenticación en SQL Server seleccionar la opción que hace referencia a esto y escribir los credenciales. Pulsar Siguiente >.



Seleccionar la opción Establecer la siguiente base de datos como predeterminada y seleccionar el nombre de la base de datos para la que se crea el origen de datos. Pulsar Siguiente.



Seleccionar otros parámetros del origen de datos según las necesidades y pulsar Finalizar.



Un mensaje de confirmación informa de todas las opciones de configuración seleccionadas durante el proceso. Es posible realizar una prueba de conexión. Pulsar Aceptar.

Tras la correcta creación del nuevo origen de datos, en la pantalla principal de administración figura en nuevo origen. Pulsar Aceptar.